ATARI

ST COMPUTER

Die Fachzeitschrift für den ATARI-ST Anwender.

Juli/August 90

DM 8,-

Ös. 64,-
Sfr. 8,-



Technobox Drafter
Konstruktives Zeichnen

BS-Handel /3
Der weißblaue Kaufmann

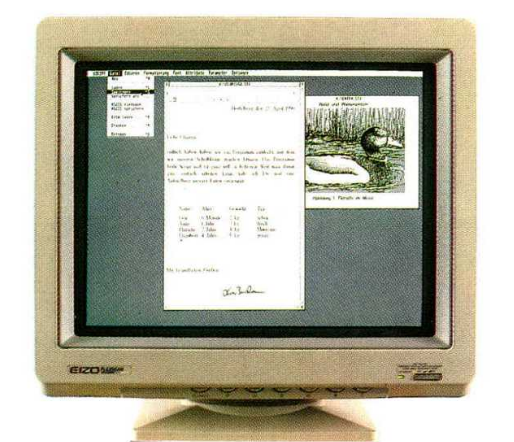
CAG
Computer Aided Graphics

Bagdad
Fernsteuerung im Netzwerk

Vortex DataJet
Eine neue Festplattengeneration?

Overscan
700 x 480 Pixel für knapp 100 DM

ATARI Messe Preview



Ob so oder auf dem SM 124, script sieht immer perfekt aus. Sowieso. Und überhaupt.

Die direkte Verbindung von Größe und Intelligenz: script.

script ist die Textverarbeitung, die sich schön und zukunftsorientiert gibt. script ist Novum und Lösung zugleich. Darum erzeugt script auch Neid, denn script kann eben einfach alles und ist ein erklärter ST-Freund, wenn mindestens 1MB vorhanden ist. Von der komfortablen Bedienung dieser Textverarbeitung einmal abgesehen, entpuppt sich script als Gigant in Verbindung mit dem Großbildschirm. script geht auch mit TT (übrigens). Was script noch alles kann, steht in dem Software-Info "script", das man bei uns anfordern kann. Den Preis verraten wir gleich: 198,--DM. Der ist ja dann eher klein.





Der Mensch als Anwender ist das Maß aller Dinge für unsere Software.



Script.
Die Textverarbeitung für Schlaue, Schnelle und Schenieser. Praktisches Handbuch, Diskette. Natürlich von der Geschmackskommission genehmigt.



HALBZEIT

Nein, ich möchte hier nicht über die Fußball-Weltmeisterschaft in Italien schreiben, sondern meine, daß die ST-Computer wieder einmal Halbzeit hat, denn diese Ausgabe ist unsere Doppelausgabe für Juli und August. Ich denke, Sie gönnen uns auch mal eine kleine Verschnaufspause, bis wir uns dann zur ATARI-Messe Ende August wiederlesen.

Natürlich werden wir in dieser Zeit nicht permanent auf der faulen Haut liegen, denn obwohl das vielgefürchtete Sommerloch seine Hochsaison feiert, kann man immer wieder hier und dort Neuigkeiten aufschnappen. Vielleicht erreicht uns sogar noch rechtzeitig ein Exemplar des bereits überfälligen TTs, der sich im Gegensatz zu der Maschine, die wir mittlerweile in der Redaktion hatten, doch noch in einigen Punkten wesentlich unterscheiden soll. Es gibt zwar diverse Gerüchte und unbestätigte Meldungen über den "neuen" TT, aber wir werden erst dann noch etwas über ihn berichten, wenn das Seriengerät bei uns auf dem Tisch steht.

Was sich bei anderen Firmen tut, haben wir Ihnen - soweit das möglich war - bereits in einem Messe-Preview zusammengefaßt. Man darf also gespannt sein, was nach dem Anpfiff zur zweiten Halbzeit die ATARI-Anwender so alles erwartet. Gelbe und rote Karten, Freistöße und Pässe in den freien Raum, alles ist drin!

Harald Egel

INHALT

SOFTWARE

GRUNDLAGEN

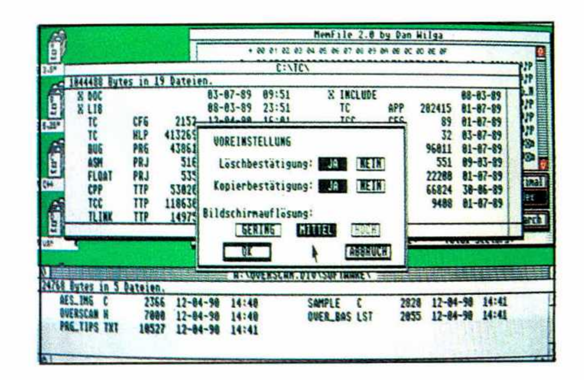
PROGRAMMIERPRAXIS



BS-Handel/3
- Kaufmann inklusive

BS-Handel ist für die ATARI-ST Computer bereits ein Software-Klassiker. Seit 1986 auf dem Markt gehört BS-Handel im Bereich integrierte kaufmännische Software sicher zu den ersten Programmen dieser Art auf dem ATARI ST. Jetzt ist die neue Version "BS-Handel/3" lieferbar. BS-Handel/3 ist keine Update-Version, sondern stellt ein komplett neu entwickeltes Programm dar. Es beinhaltet nicht nur den Funktionsumfang der Vorgängerversion, sondern wurde in der GEM-Oberfläche und bei der Verarbeitungsgeschwindigkeit völlig neu überarbeitet.

Seite **25**



Overscan
- Die Auflösungs-Erweiterung

Mit einer originellen Idee und einer einfachen Schaltung, garniert mit einem Stück Software, läßt sich die Bildschirmauflösung des ST massiv erhöhen. Das Produkt nennt sich Autoswitch-Overscan und ist auch für den kleinen Geldbeutel durchaus erschwinglich.

Seite **58**



Als Alternative, aber auch als Ergänzung zu einer RAM-Disk kann dieses Accessory eine enorme Geschwindigkeitssteigerung bei Disketten[Lese]zugriffen bewirken. Dabei wird im günstigsten Fall die Geschwindigkeit einer RAM-Disk erreicht. Die am häufigsten benutzten Sektoren werden direkt im Speicher behalten. Wenn jetzt ein Lesezugriff erfolgt, werden sie nicht mehr von Diskette, sondern aus dem Speicher geladen, was die Sache natürlich ganz enorm beschleunigt.

Seite 152

ROCKUS

© INGO STEIN MÜNCHEN 1989

Ab dieser Ausgabe wollen wir Ihnen neben den gewohnten ST-Computer-Rubriken einen neuen Wegbegleiter mit auf die Reise geben. Der Papagei Rockus soll Sie fortan daran erinnern, daß man neben der ernsthaften Computerei auch durchaus mal ein bißchen Humor bewahren sollte. In Rockus spiegelt sich der alles beherrschende Computer-Freak wieder, dem doch auch immer wieder Fehler unterlaufen und der seinen ewigen Kampf mit der Maschinen ausfechtet, die wie im richtigen Leben auch mal ihr Eigenleben entwickelt. Viel Spaß!

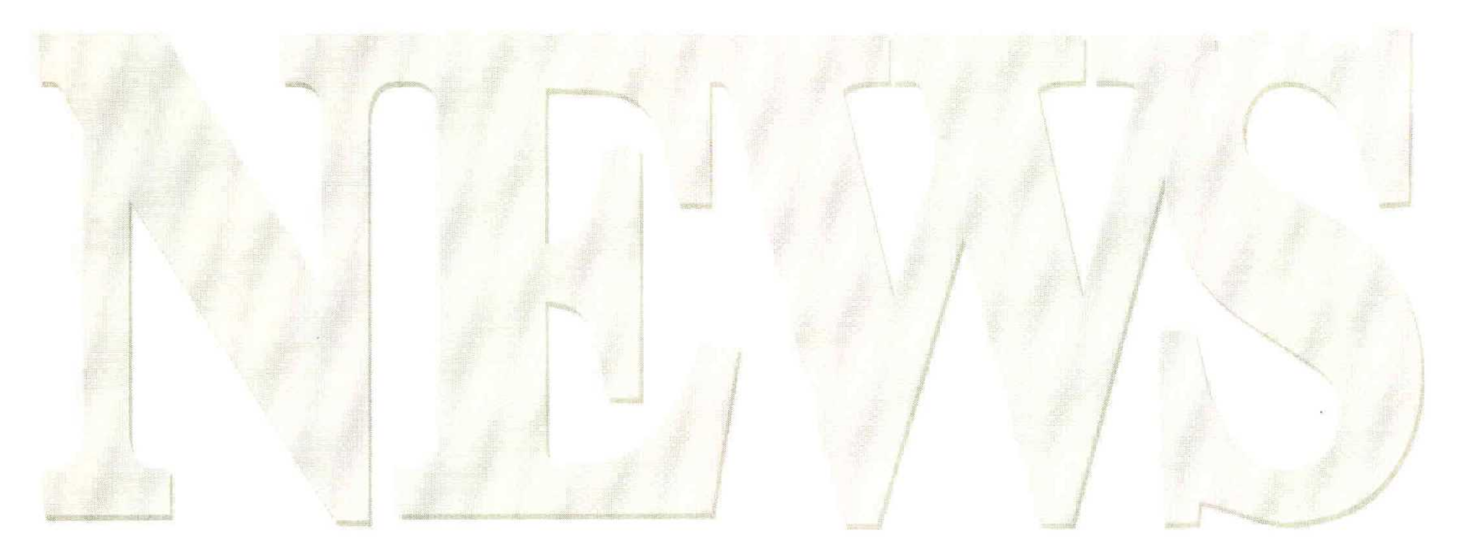
Seite 30, 44 und 109

HARDWARE

ST-Report

AKTUELLES

RUBRIKEN



Artworks-Business

Die Paderborner Firma Hesse & Herwig bietet mit Artworks-Business ein umfangreiches Gestaltungspaket für die Arbeit mit Calamus an. Artworks-Business beinhaltet ungefähr 40 copyrightfreie Briefbögen-Gestaltungen mit allen notwendigen Informationen, Markierungen, Texten, Logos, Passer- und Schnittmarken für Mehrfarbdruck, Variationsvorschläge etc.; den Briefbogen entsprechende Visitenkartengestaltungen, Nutzen zur Filmbelichtung; jede Menge Formulare in mehreren Variationen: Angebots- und Auftragsformulare für Druckereien, Rechnungs-, Mahnungs- und Lieferscheinformulare, Kursmitteilungen usw. Alle Gestaltungen und Formulare sind außerdem im umfangreichen Handbuch abgebildet. Artworks-Business erscheint zur Atari-Messe 1990. Infomaterial, Vorabinformationen und Bestellungen bei

Hesse & Herwig
Lesteweg 33
4790 Paderborn
Tel. (05251) 92231

4 MB-Grenze gesprengt

Der Mega ST wird "erwachsen". Mit einer neuen Speichererweiterung der Firma Richter Distributor aus Gevelsberg läßt sich jeder Mega ST auf 12 Megabyte aufrüsten. Es werden keine Treiber benötigt, da die 12 MB automatisch voll vom System erkannt werden. Großbildschirme laufen ohne Anpassung oder Patch! Jedes BASIC meldet einen freien RAM-Bereich von 12 MByte! Einer Dimensionierung eines Felds von 1.5 Millionen Zahlen steht also nichts mehr im Wege. Die erforderliche Grundausbaustufe von 4 auf 6 MB kostet DM 1598,-, die Vollausbaustufe inklusive Grundausbaustufe DM 3998,-. Für 6 MB auf 8 MB sind DM 998,- zu berappen, 6 auf 10 MB kosten DM 1798,-.

H. Richter Distributor
Hagener Straße 65
5820 Gevelsberg
Tel. (02332) 2706

Neue Modems von Schewe

Modems mit einem Jahr Garantie und 14 Tage Rückgaberecht sind jetzt bei der Hamburger Firma Carl Schewe erhältlich. Ein 9600 Baud-Modem mit MNP5 ist beispielsweise für DM 1998,- erhältlich. Ein 4800 Baud Telefax-Modem mit integriertem 2400 Baud-Modem kostet dagegen nur DM 399,-, ein Tornado 2400 Baud-Modem ohne FAX ist bereits ab DM 298,- erhältlich. Für das Maxmodem 2400E/M5 mit MNP5 müssen dagegen schon DM 599,- berappt werden.

Carl Schewe GmbH & Co.
Essener Straße 97
2000 Hamburg 62
Tel. (040) 5270321

Save It Memory und STAN

Mit Hilfe von SIM läßt sich der komplette Speicher des ST auf Diskette sichern. Der auf Disk gesicherte Hauptspeicher kann jederzeit wieder geladen werden. Mit diesem Programm können Sie Ihre Arbeit am ST jederzeit unterbrechen und Tage danach an genau der selben Stelle wieder fortfahren. Mit STAN-ST lassen sich umfangreiche Statistiken erstellen. Benötigt wird allerdings ein ST mit mindestens 1 MB Speicher. Informationen und Bestellungen für die beiden Programme sind erhältlich bei

Software & EDV-Beratung Weber
Nelkenstraße 6
8508 Wendelstein 2

TommySoftware legalisiert Raubkopien

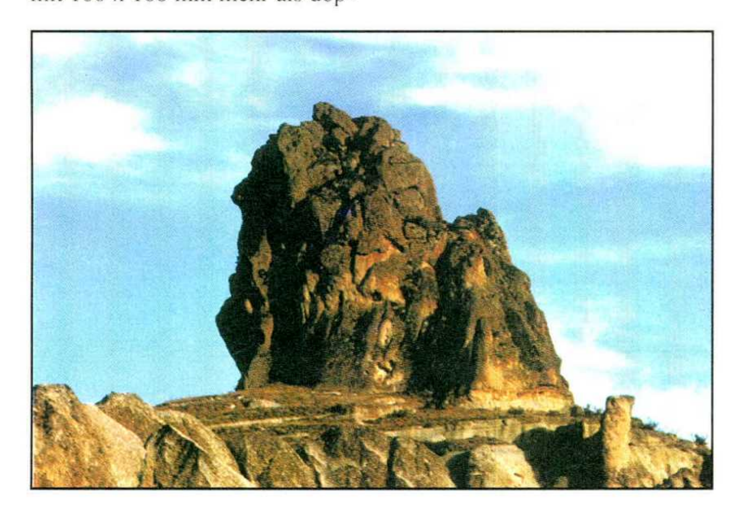
Das Unternehmen TommySoftware aus Berlin legalisiert mit Wirkung vom 01.07.1990 sämtliche Raubkopien seiner Software-Produkte in der DDR. Mit diesem Schritt will der Software-Hersteller allen Anwendern in der DDR die Möglichkeit bieten, die Benutzung seiner Software zu legalisieren. Voraussetzung ist der kostenlose Erwerb einer Registrationsnummer bei dem Unternehmen im Schriftverkehr. Die Aktion ist zeitlich bis zum 30.09.1990 begrenzt. Alle später eingehenden Registrationen können leider nicht mehr berücksichtigt werden. In der DDR gibt es einen unübersichtlichen Schwarzmarkt für Raubkopien. TommySoftware strebt mit diesem unternehmenspolitischen Schritt die Sicherstellung seiner Interessen in der DDR an.

TommySoftware
Selchower Straße 32
1000 Berlin 44
Tel. (030) 621406-3

Farbhandyscanner für den ST

Die TmS GmbH Regensburg bietet ab sofort Adaptionen für den ST für die Handyscanner Sharp JX 100 sowie Epson GT 1000 an. Die Bilddaten beider Scanner werden mittels der beiliegenden Software mit bis zu 256 Graustufen bzw. 16,8 Mio. Farben bearbeitet. Die Ausgabeauflösung des Epson GT 1000 reicht bis zu 200 dpi, die des Sharp JX 100 bis 400 dpi. Beide Scanner tragen zwar die Bezeichnung Handyscanner, jedoch werden sie nicht über die Vorlage gezogen, sondern nur auf die Vorlage gelegt. Sodann tasten beide Geräte die Vorlage berührungslos ab, wodurch ein hoher Bedienungskomfort erreicht wird. Die maximale Scan-Fläche des Epson GT 1000 beträgt 74 x 105 mm, die des Sharp JX 100 ist mit 100 x 160 mm mehr als doppelt so groß. Beide Scanner können auch Durchsichtvorlagen einscannen, wodurch sie Verwendung als Dia-Scanner finden können. Als Hardware-Voraussetzung reicht ein ST mit 1 MB Speicher sowie ein Sm 124 bzw. zusätzlich ein Farbschirm für die Auflösung von 320 x 200 in 16 Farben. Für alle, die ihren Scanner bis zuletzt ausreizen wollen, steht auch ein Scanmodul für das Bildverarbeitungssystem TmS Cranach zur Verfügung. Der Preis für den Epson GT 1000 inklusive umfassender Bildverarbeitungssoftware beträgt DM 1798,-, für den Sharp JX 100 DM 2498,-.

TmS GmbH
Cranachweg 4
8400 Regensburg
Tel. (0941) 95163



Umrüstsätze von Compu-Ware

Die Firma Compu-Ware hat ihr Lieferprogramm erweitert. Ab sofort sind auch HD-Laufwerke erhältlich. Hierbei handelt es sich um eine Eigenentwicklung der Firma Compu-Ware. Angeboten werden Umrüstsätze für die Mega ST-Reihe sowie Einzellaufwerke im 3,5"- und 5,25"-Diskettenformat. Ebenso wird eine Doppelstation mit beiden Laufwerkstypen angeboten. Zum Lieferumfang gehört ein Formatierprogramm sowie eine Einbauplatine für die 8- bzw. 16 MHz-Umschaltung. Somit stehen dem Anwender bis zu 1,6 MByte Speicherplatz auf 3,5"-HD-Disketten zur Verfügung. Auf Wunsch wird die externe Diskettenstation auch mit intelligenter Anzeige geliefert. Preisangaben zu den Projekten erhalten Sie von der Firma Compu-Ware. Für die Festplattenbenutzer mit großem Datenvolumen bietet Compu-Ware ein neues Programm an. HD-SEARCH ist ein rationelles und komfortables Utility für die Festplatte. Dier Leistungsmerkmale sind: schnelles Aufrufen einer Partition mit sofortiger Anzeige des Inhalts, Inhaltsanzeige bestimmter Ordner, volle Pfgadanzeige und vieles mehr.

Compu-Ware
Dreufte 5
4250 Bottop 2
Tel. (02045) 6302



Alternative Maus

Von Logitech kommt jetzt eine optomechanische Maus mit zwei Tasten, die eigens für den Atari ST entwickelt wurde. Die Standardauflösung des Geräts beträgt 200 dpi, doch die Cursor-Reaktion läßt sich über das Schreibtischzubehör "Pilot Control" individuell einstellen. Logitech liefert seine Pilot-Maus für den ST mit einer Gratiskopie des Geschicklichkeitsspiels "Pipe Mania" (Kurzfassung mit 3 Leveln) aus. Dieses Programm wurde vom Game Player-Magazin als Spiel des Jahres ausgezeichnet. Die Pilot-Maus ist in einer deutschen, englischen, französischen und italienischen Version zum Preis von DM 99,- erhältlich bei allen Logitech-Händlern. Weitere Informationen gibt's bei

LOGI Computer & Software
Vertriebs GmbH
Tel. (089) 1784061

MegaPlot

MegaPlot ist speziell für den Atari ST zur grafischen Darstellung von Meßwerten entwickelt worden. Es können bis zu 20 verschiedene Meßreihen bearbeitet werden, die Anzahl der Meßwerte wird nur durch den verfügbaren Speicherplatz begrenzt. Das Programm kann die Meßpunkte wahlweise durch Linien verbinden oder Funktionen berechnen (Polynome 2. bis 5. Ordnung, e-, exponentielle- oder logarithmische Funktionen, Spline und Gätterung von Meßwerten); auch ist die Darstellung als Blockdiagramm implementiert. Die Achsen lassen sich linear, einfach- oder doppelt-logarithmisch skalieren. Weiterhin können zu allen Meßpunkten Fehlerbreiten eingezeichnet werden; die Meßwerte lassen sich durch mathematische Operationen (u.a. Multiplikation, Addition und Kehrwertberechnung) verändern. Auch das Vertauschen von Meßreihen sowie der Datenimport über ein ASCII-File ist problemlos möglich. Die Programmsteuerung ist sowohl über Pull-Down-Menüs als auch über Tastaturbefehle möglich, mittels der HELP-Taste lassen sich alle Tastaturbefehle und deren Bedeutung anzeigen. Die wichtigsten Operationen lassen sich über die Funktionstasten F1-F10 durchführen. MegaPlot benötigt mindestens 1MB Speicher und einen monochromen Monitor und ist unter allen TOS-Versionen lauffähig. Das Programm wird bis zum 31.8.1990 zum Einführungspreis von DM 169,- (später DM 189,-) angeboten, eine Demoversion ist für DM 10,- erhältlich.

WBW-Service
Sielwall 87
2800 Bremen 1
Tel. (0241) 75116

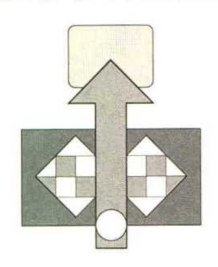
BELA TIPS

G+PLUS

G+Plus ist ein vollständiger Ersatz für GDOS. Fonts und Gerätetreiber werden ohne Neustart des Rechners bei Aufruf eines Programms automatisch nachgeladen!

G+PLUS -
Der GDOS-Ersatz
Unverbindliche Preisempfehlung
DM 79,-

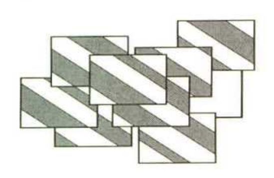
HOTWIRE



Die Shell für den geplagten ST-Anwender. HotWire startet jedes ST-Programm durch Drücken einer Taste, egal wo das Programm steht! Hotwire erlaubt dazu über 400 Tastenkombinationen.

HOTWIRE -
Die Starthilfe
Unverbindliche Preisempfehlung
DM 79,-

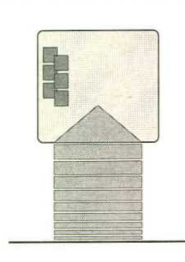
REVOLVER



Der Profi-Switcher für Ihren ATARI ST. Wo andere Programme den Dienst quittieren, da bietet REVOLVER Sicherheit. Resetfest in jedem Rechnerteil und mit umfangreichen Utilityfunktionen ist REVOLVER ideal für Programmierer, Musiker und Anwender, die mehr aus ihrem ATARI ST machen wollen.

REVOLVER -
Der Profi-Switcher
Unverbindliche Preisempfehlung
DM 79,-

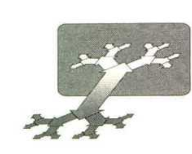
MULTIDESK



Multidesk lädt bis zu 32 .ACC-Programme nach und verbraucht nur einen Eintrag im DESK-Menü. Das Nachladen ist jederzeit möglich! Oder starten Sie jedes Accessory wie ein normales Programm. MultiDesk machts möglich!

MULTIDESK -
Der ACC-Manager
Unverbindliche Preisempfehlung
DM 79,-

XBoot



XBoot ist die Lösung für den gestreßten Festplattenbesitzer. Es ermöglicht bei jedem Boot-Vorgang die Auswahl der zu ladenden Accessories und AUTO-Ordner-Programme. So nutzen Sie Ihren ST optimal und verschwenden keinen Speicherplatz durch nicht benötigte residente Programme. Ganz nebenbei lassen sich beliebige GEM-Programme automatisch starten - auch mit den alten TOS-Versionen 1.0 und 1.2!

Doch damit nicht genug: Um nicht bei jedem Booten von der Festplatte sämtliche Einstellungen erneut vornehmen zu müssen, kann für alle Anwendungen ein SET definiert werden. Dadurch beschränkt sich das Ändern der Arbeitsumgebung auf einen einfachen Mausklick oder Tastendruck.

XBoot macht das Unmögliche möglich. Es ist das erste (AUTO-Ordner) Programm für den ST, das den Komfort eines GEM-Programms mit kompletter Mausbedienung bietet.

XBoot
Der Boot-Manager
Unverbindliche Preisempfehlung
DM 69.-

CADJA

Komplexe technische Zeichnungen sind immer eine langwierige Arbeit, und wenn sie einmal fertig sind, will man nichts mehr von Änderung oder Korrektur wissen. Sind diese aber doch notwendig, beginnt das Kratzen oder man darf neu zeichnen. Warum das alles noch, wenn jetzt CADJA da ist?
Mit CADJA zeichnen Sie wie Sie es gelernt haben - nur um einiges schneller und bequemer. Und falls Sie noch eine Korrektur wünschen, ist die Zeichnung jederzeit vorhanden und Sie können sie aufrufen, korrigieren und neu speichern. Nach den Ausgabegeräten fragen Sie? CADJA kann fast alles: Nadeldrucker, Laserdrucker, Tintenstrahl, aber auch Plotter und Plotter-Sprache.
Was das Programm außerdem noch kann? Ja, das läßt sich in diesem Rahmen nur schwer beantworten. Fordern Sie einfach unser CADJA-Prospekt an.

TURBO ST

Jetzt in der Version 1.8.

Die Lösung: Laden statt löten! Machen Sie Ihrem ATARI ST "Beine" und beschleunigen Sie die Anzeige von Texten auf dem Bildschirm auch ohne Blitterchip. Turbo ST ist sogar noch schneller als der Blitter und damit besonders interessant für Besitzer der STs, in denen der Blitter überhaupt nicht vorgesehen ist.

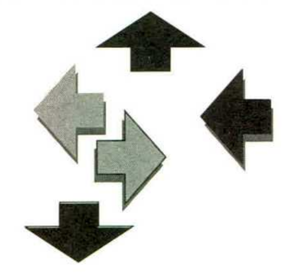
TURBO ST -
Der Softwareblitter
Unverbindliche Preisempfehlung
DM 89,-

Dieser Platz ist sehr wichtig!

Hier könnte in Zukunft Ihr Programm stehen. Schreiben Sie uns oder schicken Sie uns ein Demo Ihres Programms.

BELA Computer
Unterortstraße 23-25
6236 Eschborn

INTERLINK ST



INTERLINK ST ist das komfortabelste DFÜ-Programm für den ATARI ST und damit ideal für den Einsteiger und den Profi. So urteilen zumindest die Besitzer, die die Kommunikation und den weltweiten Datenaustausch mit Hilfe von INTERLINK ST nicht mehr missen möchten. Wann gehen Sie auf die Datenreise?

INTERLINK ST -
DFÜ im Griff
Unverbindliche Preisempfehlung
DM 79,-

SALDO

SALDO ist ein Programm, das Ihnen erlaubt, die Kontrolle Ihrer Finanzen in den Griff zu bekommen. Sie können SALDO für private Zwecke, aber genausogut als Einnahme- und Überschußverwaltung für die gewerbliche Tätigkeit einsetzen. SALDO bietet mit seiner Vielzahl an Funktionen alle nur denkbaren Möglichkeiten, die eingegebenen Daten zu manipulieren. Sie können z.B. sortiert oder aufgesplittert nach verschiedenen Kriterien auf dem Bildschirm dargestellt oder auf dem Drucker ausgegeben werden.
Es würde einfach zuviel, hier jedes einzelne Detail von SALDO aufzuzählen - man muß es gesehen haben.

SALDO
Unverbindliche Preisempfehlung
DM 79.-



BELA Computer Layout- und Vertriebs GmbH Unterortstr. 23-25 D-6236 Eschborn Tel: 06196/481944 Fax: 481930

PREVIEW

ATARI-Messe 1990

Was gibt's zur Atari-Messe Neues? Diese Frage werden sich viele Anwender bereits gestellt haben. Wir präsentieren Ihnen schon jetzt eine Vorschau der Neuigkeiten, die Firmen für und rund um den Atari ST anbieten wollen.

Die Organisation und der Aufbau der Messe dürften auch recht interessant sein. Deshalb hier einige Informationen von Atari selbst. Seit 1989 findet die Atari-Messe zusammen mit der "Aktiv Leben" (größte Verbraucherausstellung in Düsseldorf und Umgebung) statt. Die Aktiv Leben '90 läuft vom 18. bis 26.08.1990 in den Hallen 7 bis 10, die Atari-Messe vom 24. bis 26.08.1990 in den Hallen 11 und 12. Die Eintrittspreise sind für beide Veranstaltungen gültig: Erwachsene zahlen DM 7,-, Schüler und Studenten nur DM 4,-. Als Besonderheit wird eine Kombi-Karte für die Messen zu den o.g. Preisen angeboten, dafür hat dann jeder Besucher die Möglichkeit, eine kostenlose Rückfahrt der Stufe VRR 5 (Radius von zirka 40 km um Düsseldorf) und Eintritt in Anspruch zu nehmen. Die Kombikarten gibt es an allen VRR-Vorverkaufsstellen.

Am Atari-Zentralstand in Halle 12 sind insgesamt vier "Arbeitstischmodule" sowie ein Portfolio-Tisch und ein "Business Managing Center" geplant. Die Neuheiten selbst werden erst kurz vor der Messe bekanntgegeben. An Tisch 1 sind Unix-TT und eine ATW zu sehen. Tisch 2 wird mit TT-Geräten ausgestattet. An Tisch 3 können sich alle PC-Freunde ihrer Geräte ergötzen. Tisch 4 bietet einen Mischmasch aus Stacy, CD-ROM und anderen Geräten. An einem sparaten Tisch werden nur Portfolios gezeigt. Der letzte Tisch des Zentralstands nennt sich "Business Managing Center". Hier wird das Programm BSS Plus inklusive Atari-Net an 5 Arbeitsplätzen hintereinander gezeigt.

In Halle 12 findet sich auch ein DTP-Center. Auf 650 m² wird gezeigt, wie man von der Idee über Layouts zur farbigen Vorlage kommt. Unter anderem kommen zur Vorführung: Inline-Präsentationswand, 3K-Präsentationswand, Ausstellung "Page Wettbewerb", Werbebüro durch Frenzel & Partner, Redaktion für Messezeitung, GMA-Plotter, Satzbelichtung / Druckmaschine und Workshop-Raum für Grafiker etc.

	1987	1988	1989	1990 (ca.)
Aussteller	79	112	155	200
Besucher	20000	26000	35000	50000
Fläche (m²)	4000	8000	12000	20000

Die Daten der vergangenen umd kommenden Atari-Messen

Im Forum in Halle 12 halten verschiedene Referenten Vorträge. Professor Dr. Siemsen von der FH Ostfriesland und Gary Lawmann von Atari England berichten hier von Transputern. Henry Plummer von Atari USA versucht, den Zuhörern Unix näherzubringen. Bei einem Wettbewerb wird hier in Zusammenarbeit mit der Red. PC Praxis die beste Software für den Atari Portfolio gesucht. Technobox bietet allen Technobox-Programmbenutzern ein Anwenderforum mit anschließender Diskussion. Auch im Rahmen verschiedener Podiumsdiskussionen werden umfangreiche Themengebiete angeboten. Frau Dr. Cillie Rentmeister spricht im Forum ein kontroverses Thema an: Frauen und Computer. Weitere Themen des Forums sind: Computer in der Schule, eine Calamus-Präsentation mit Podiumsdiskussion, die technischen Entwicklungen bei Atari in den 90ern, Computer und Kreativität und das große Atari-Messegewinnspiel.

Im Rahmen eines Themas, das wir erst durch die ST Computer publik gemacht haben, hat jetzt endlich auch Atari Interesse gezeigt. Unter dem Titel "Atari-Computer in Bildung und Wissenschaft" stellen verschiedene Gruppen aus. Die Atari-Interessengemeinschaft mit Herrn Schulte-Borberg, über die wir bereits mehrfach berichteten, ist ebenso vertreten wie Abgesandte anderer Institutionen: Heinrich-von-Kleist-Gymnasium Bochum, die Herren Karl, Höhne und Rosseau aus Bayern, Herr Pulver aus Rheinland-Pfalz, Herr Deetz aus dem Saarland, die Flugsicherung der FH Düsseldorf, das Institut für Didaktik der Geographie der Universität Frankfurt, das Institut für Angewandte Mathematik der Universität Karlsruhe, Herr Gründelmann aus der DDR (Randelemente der BEM) sowie die Heinrich-Hertz-Kollegschule aus Düsseldorf (CAE). Diese Aussteller belegen insgesamt 180 m².

Halle 11 dürfte für Spiele-Freaks und Grafik-Enthusiasten ebenso interessant sein wie für die Sound-Fanatiker. Hier werden das gute, alte VCS2600 und das ebenfalls alte 7800-Telespiel "vorgestellt". Schwerpunkt dürfte jedoch das neue Hend-Held-Spiel Lynx sein. In einem Wettbewerb sollen die besten "Lynxer" der Welt ermittelt werden.

In einer Computergalerie, ebenfalls in Halle 11, werden 100 Computergrafiken von Prof. Dr. Georg Nees gezeigt, die mit einem Atari ST erstellt wurden. Die Leitung der Präsentation hat Dr. Gerhard Dotzler inne ("computer art fascination").

Wie auch in Halle 12 existiert auch in Halle 11 ein Forum. Hier will man den Interessenten unter der Leitung von Prof. Enders das Musiklernen mit dem Computer näherbringen. Musikprogramme en masse werden den geneigten Musikern "aufs Ohr gedrückt": C-Lab, Steinberg und Hybrid Arts kommen hier ebenso nicht zu kurz wie die Hotz-Box, die bereits auf der letzten Atari-Messe zur Vorstellung kam. Zum krönenden Abschluß berichtet Prof. Dr. Georg Nees über "25 Jahre künstliche Kunst".

Was haben nun die Firmen selbst an Neuigkeiten zu bieten? Wir befragten einige von ihnen, um Ihnen schon jetzt einen kleinen Überblick bieten zu können. Was dabei herausgekommen ist, lesen Sie in den nächsten Zeilen.

Bavaria-Soft

Hier geht man mit der Zeit - Die Programme BS-Fibu, BS-Handel und BSS+ erhalten alle eine neue Versionsnummer: 3.0 - man darf also auf die Neuerungen gespannt sein! Das Programm BSS+ wird sogar netzwerkfähig.

Biodata

Das bekannte BioNet, das von der Flächenausdehnung her zu den 100 größten Ethernet-Netzwerken der Welt gehört, erhält ab sofort auch einen Server unter UNIX. Weiterhin können nun auch Nodes auf dem Macintosh betrieben werden. Damit läuft das Netzwerk zusammen mit ST, PC, UNIX und MS-DOS.

CASH

TiM II erhält eine DATEV-Schnittstelle namens TDS. TDS erlaubt erstmalig auf dem Atari ST, Daten einer Buchführung der "Time is Money"-Serie in ein Diskettenformat zu bringen, welches ein Steuerberater mit DATEV-PC und der DATEV-Schnittstellen-Software (DSSW) lesen kann. TDS ist bereits ab dem 1.8.90 im Handel. Demodisketten können ab 1.7.90 für DM 30,- bei CASH bezogen werden.

DMC

Lange erwartet: Calamus SL soll zur Atari-Messe endlich fertig sein. Es wurde bereits in einer ersten Version auf der CeBIT vorgestellt. Das Programm unterstützt Farbe und Modultechnik. Neu ab Oktober ist auch Calamus S. Diese Version des DTP-Programms unterstützt keine Modultechnik und hat keine Farbfähigkeit, ansonsten aber alle Neuerungen des Calamus SL.

3K

Retouche Professional wird bereits in der 2. Juniwoche ausgeliefert, auf der Atari-Messe aber zum ersten Mal der breiten Öffentlichkeit komplett vorgestellt. Retouche Professional ist ein Bildverarbeitungsprogramm für, wie der Name schon sagt, Profis.

Drews

Die Drews EDV+BTX GmbH hat für die Version 4.0 des BTX-Managers eine Postzulassung beantragt. Der BTX-Manager 4.0 soll 16 Graustufen im monochromen Modus und 16 Farben im Farbmodus darstellen. Sobald die 4.0-Version erhältlich ist, löst sie automatisch die Version 3.0 ab. Außerdem wird das Programm eine Preissenkung erfahren. Man darf gespannt sein.

IFA-Köln

James 3.0 wird zwar schon verkauft, auf der Atari-Messe aber zum ersten Mal der breiten Öffentlichkeit vorgestellt. Es bietet jetzt logarithmische Darstellung, Momentum-, Volumen- und 5-Jahres-Charts sowie viele andere Neuerungen.

LogiLex

Ein völlig neues Programm kommt von LogiLex. Mit 1st_Lock läßt sich ein perfekter Datenschutz für Disketten und Festplatten einrichten. Daten, die abgespeichert werden sollen, werden noch vor dem Schreiben codiert und so wirkungsvoll vor unbefugtem Zugriff geschützt. Decodiert werden sie erst wieder direkt nach dem Einlesen. Eine Version für MS-DOS-Geräte ist ebenfalls erhältlich.

novoPLAN

1st_FibuMAN ist eine Einsteigerbuchführung. Sie erscheint zur Atari-Messe, außerdem FibuMAN in der Euroversion 4.0. 1st_Fibuman läßt 199 Buchungen pro Monat zu. Die Schnittstellen sind bei beiden Programmen dBASE-kompatibel. Weiterhin existiert eine ASCII-Schnittstelle, die den vielfältigen Im- und Export von Daten zuläßt. Ob die von novoPLAN geplante Fakturierung zur Atari-Messe fertig wird, steht noch nicht fest.

rhothron

Für diejenigen, die das Gehäuse des TT nicht mögen, bringt die Firma rhothron ihr eigenes auf den Markt und dürfte damit einen reißenden Absatz finden. Weitere Neuigkeiten für den TT sind eine Steckplatzerweiterung (8 oder 20 VME-Bus-Schnittstellen), eine IEEE 488, eine Ethernet-, eine Farbgrafikkarte mit 256 Farben bei einer Auflösung von 640x480 Pixeln sowie ein A/D- bzw. D/A-Wandler. Weiterhin kommen Festplatten für den TT zur Vorstellung. Auch neue Software für den ST ist erhältlich: Bilder und Grafiken können mit Meßwerten belebt werden. Man darf auch hier gespannt sein.

Computerware

Unter dem Namen "CLImax" stellt die Firma Computerware aus Köln einen CLI vor, dessen Bedienung etwas an MS-DOS erinnert. Genau das richtige für Tastenakrobaten!

Eickmann Computer

Die Frankfurter Festplattenschmiede Eickmann Computer erweitert ihr Angebot. Ab der Atari-Messe soll auch ein preiswertes Netzwerk im Angebot sein. Das System soll DM 500,- pro Rechner kosten und Übertragungsraten von 100 kB/Sekunde aufweisen. Sollten diese Werte stimmen, dürften einige andere Netzwerke arg ins Trudeln geraten. Turbo 16, der ST-Beschleuniger, kommt jetzt auch als Turbo 16 II auf den Markt. Dieses Gerät hat einen Coprozessor eingebaut. Außerdem soll auch ein Gerät namens Turbo 30 auf den Markt geworfen werden, das den ST in die Lage versetzt, mit einem 68030-Prozessor mit 16 oder 32 MHz Taktfrequenz zu arbeiten. Für kurz nach der Atari-Messe sind auch neue Festplatten angekündigt: kleinste Gehäuse, die nur wenig Platz beanspruchen, leise und schnelle SCSI-Platten, Geräteadressen von außen verstellbar - so lauten die ersten technischen Vorabinformationen von Eickmann.

SciLab

Das Programm SciGraph erscheint zur Atari-Messe in einer neuen Version. Mit dem neuen Produkt sind auch 3D-Rotationen möglich. Außerdem kann das erweiterte SciGraph durch die Benutzung von Metafiles auch True-Color-Berechnungen durchführen. Dazu gehört auch, daß sich der Fluchtpunkt und die Beleuchtung frei einstellen lassen. Der Grafik-Editor kann ab der Messe auch mit Vektor-Fonts



FÜR IHREN ATARI ST
(Mega – 1040 – 520 – 260)
TOWER
POWER
Wenn Sie vor lauter Computer keinen Platz mehr auf dem Schreibtisch haben. Wenn sie der Gerätelärm beim Arbeiten stört oder wenn es Sie ärgert, daß so viel Einzelgeräte herumstehen, dann braucht Ihr ST – TOWER POWER –
IDEAL FÜR HARDWARE TUNING!

und Bézier-Kurven arbeiten. Gleichzeitig soll auch eine MS-DOS-Version des Programms erscheinen. Durch die Neuerungen sind auch Grafiken aus GEM Artline übernehmbar. Als "kleine Zugabe" wird außerdem ein Programm zur Umwandlung von Vektor- auf Bitmap-Grafiken angeboten.

Technobox

Bei der samstäglichen Atari-Messe bietet Technobox allen Technobox-CAD-Besitzern ein Forum an, auf dem Erfahrungen ausgetauscht und interessante Neuigkeiten erfahren werden können. Die Veranstaltung nennt sich "Technobox-Forum". Mehrere hundert Benutzer haben sich bereits angekündigt. Natürlich werden auch neue Programme bzw. Updates/Upgrades gezeigt. Der Technobox-Drafter erscheint in einer neuen Version, die Assoziativität bei der Bemaßung erlaubt. Auch in die Koordinateneingabe wurde entscheidend eingegriffen, sie ist ab der Messe wesentlich komfortabler. Als besonderes Bonbon wird in den Drafter eine Kommunikationsschnittstelle eingebaut, mit der sich das Programm komplett von außen steuern läßt. Das allerdings nicht durch eine eigene Pipeline, sondern komplett über GEM. Dadurch müßte der Drafter dann auch problemlos auf dem TT funktionieren. Eine erfreuliche Meldung: Trotz der vielen Neuerungen bleiben die Preise stabil - Technobox CAD/II kostet weiterhin DM 1998,-.

CCD

Das Warten auf Tempus Word 1.0 soll auf der Atari-Messe endlich ein Ende haben. Tempus Word 1.0 soll hier zwar vorgeführt, jedoch nicht verkauft werden - die Version muß erst noch Tests überstehen. Weitere Neuerungen finden sich auch bei der "normalen" Version von Tempus, die auf der Messe in der Version 2.10 erscheinen soll. Tempus 2.10 ist an Großbildschirme angepaßt, hat den Fileselectorbox-Fehler mit TOS 1.4 behoben (der Balken sitzt wieder an der richtigen Stelle) und bearbeitet nun endlich auch echte Tabulatoren!

Bela

Die Eschborner Firma Bela wartet gleich mit mehreren Neuigkeiten auf. Summa, ein neues Programm, ist eine Branchenlösung für Handwerker. Lagerverwaltung, Buchhaltung und alles, was Handwerker benötigen, ist in diesem Programm integriert. Mit XBoot, ebenfalls einem neuen Programm, stellt Bela ein Boot-Programm vor. Mehrere Benutzerprofile, Autostartprogramme, steuerbar per Maus, obwohl direkt nach dem Kaltstart noch kein GEM installiert ist. Die nächste Neuvorstellung nennt sich Saldo und ist zur Kontrolle der Finanzen des Benutzers bestimmt. Durch ein komfortables Abfragsystem ist der Datenbestand nach verschiedenen Kriterien sortierbar. Ein komplett neues Programm, das bisher nur vom Macintosh bekannt ist, ist ein Makro-Editor. Dadurch lassen sich immer wiederkehrende Abläufe automatisieren. So läßt sich beispielsweise die Maus durch den Makro-Editor steuern, aber auch Tastendrücke können gespeichert werden. Auf der Messe wird außerdem mit dem CAM-Teil des Programms CADja eine Fräsmaschine demonstriert. Weiterhin geplant sind bei Bela auch ein Backup-Programm für Festplatten sowie einige andere nützliche Utilities, die das schwere Leben mit dem ST etwas erleichtern sollen.

Weide

Die bekannte Firma Weide bietet neuerdings auch Programme an. So wurde schon auf der CeBIT ein Programm namens "Adress Direkt" vorgestellt. Dieses imposante Lager-, Fakturierung- und Adreßprogramm, das auf allen Großbildschirmen läuft und komplett per Tastatur steuerbar ist, wird auf der Atari-Messe in einer wesentlich erweiterten Version vorgestellt werden. Ganz neu ist auch ein Kopierprogramm über DFÜ. Dateien oder gar ganze Partitionen sollen sich mit diesem Programm über die parallele oder serielle Schnittstelle kopieren lassen. Wird beim Backup auf eine andere Platte ein bereits vorhandenes Programm gefunden, kann es in einen Backup-Ordner gesichert werden, bevor es überschrieben wird. Das Programm könnte eine große Erleichterung für Programmierergruppen sein, die auf mehreren Rechnern arbeiten - die Messe wird es zeigen. Weiterhin geplant ist auch eine C-Library-Dokumentation, mit der die Analyse von C-Sources problemlos möglich sein soll.

MAXON Computer

Lange erwartet, endlich verfügbar: MACH 16, der Beschleuniger aus dem Hardware-Sonderheft, wird ab der Atari-Messe erhältlich sein. Optional kann ein 68881-Prozessor eingebaut werden. Außerdem läßt sich die MGE-Grafikkarte ab sofort auch mit 16,7 Millionen Farben ausrüsten, 256 davon können gleichzeitig benutzt werden. Natürlich ist es dann auch möglich, 256 echte Graustufen (!) zu benutzen! Die Erweiterung wird möglich durch eine neue Color-Lookup-Table, die für DM 79,- inklusive Software nachgerüstet werden kann. Eine weitere Neuerung, die für alle Mega ST-Besitzer interessant sein dürfte, ist eine voll kompatible 68020-Karte. Ihre Daten: 32 Kilobyte, 32 Bit 0 Waitstate Cache, dreieinhalbfache Geschwindigkeit im Vergleich zum 68000/8 MHz in der Praxis.

TmS

TmS Vektor wird ab der Atari-Messe in der Version 2.5 ausgeliefert. Diese neue Version unterstützt dann auch Bézierkurven. Eine weitere Neuigkeit ist auch das Programm Cranach Studio. Es ist angepaßt worden an den Sharp- und Epson GT1000-Handyscanner. Bei der Bildverarbeitung können 6 Lassos gleichzeitig benutzt werden. Durch eine echte virtuelle Speicherverwaltung ist die Bildgröße außerdem ab sofort unbeschränkt. Verschiedene Filter und ein Zooming fehlen natürlich ebensowenig wie eine Farbseparation. Das Programm besitzt eine Calamus-ähnliche Benutzeroberfläche und ist modular aufgebaut, d.h. Zusatzprogramme können als Module eingebunden werden. Cranach Studio arbeitet als eines der ersten Programme auch mit der Chili-Grafikkarte der Gevelsberger Firma Richter Distributor zusammen. Außderdem sind Anpassungen an den Mitsubishi-Farbscanner vorgenommen worden. Cranach Studio wird auch in einer speziellen Version für Maler vertrieben werden. Das Programm soll laut TmS alle bisher erschienenen Bildverarbeitungssysteme für den ST und den Mac (!) übertreffen. Man darf also gespannt sein!

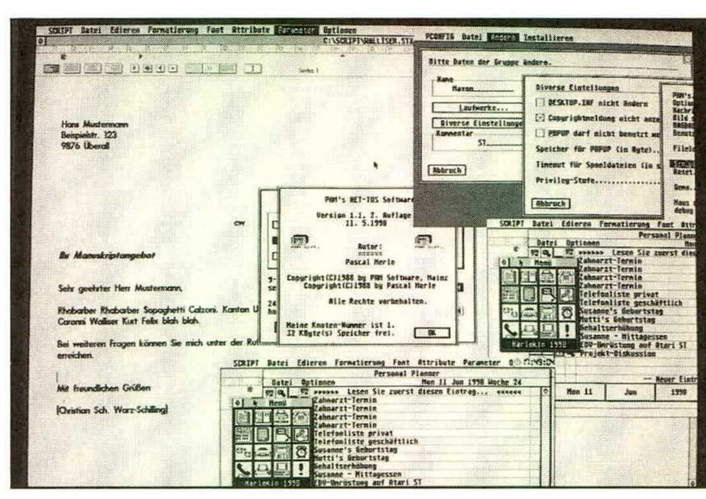
ICD

Die bekannten Host-Adapter der Firma ICD bekommen einen neuen Bruder. Die Firma plant, einen High-End-Host-Adapter für gehobene Ansprüche auf den Markt zu bringen. Das Gerät ist für den Industrieeinsatz entwickelt worden und unterstützt Arbitration, d.h. es kann prinzipiell auch ohne den ST arbeiten bzw. andere Adapter von sich aus ansprechen. Eine weitere Neuerung, die sich hervorragend in die Reihe der bereits erschienenen ST-Beschleuniger einreiht, ist ein "Accelerator", der den ST auf eine Taktrate von höchstwahrscheinlich 16 MHz hochbohren wird. Ob ein Cache eingebaut sein wird, stand bei Redaktionsschluß noch nicht fest.

MP

Bagdad

Fernsteuerung im Netzwerk



Bagdad in Aktion: Hier werden vier STs gleichzeitig gesteuert!

Was auf dem Macintosh möglich ist, sollte eigentlich auch mit dem ST zu verwirklichen sein. Timbuktu ist das Programm für den Mac, Bagdad nennt sich das Programm, das die Fernsteuerung eines ST von einem anderen aus ermöglicht.

Um den ST fernsteuern zu können, müssen verschiedene Komponenten vorhanden sein. Da ist zum ersten ein Netzwerk, in diesem Fall PAMs NET (Test in Ausgabe 10/89). Weiterhin wird natürlich mindestens ein zweiter ST benötigt. Damit wäre die Fernsteuerung zwar prinzipiell schon möglich, jedoch nur sehr unkomfortabel - man würde immer nur einen Bildschirm sehen. Mit einer Grafikkarte und einem größeren Monitor hätte man alles, was man braucht, auf einen Blick auf dem Monitor. Konkret braucht man zur Fernsteuerung also PAMs NET, mehrere ST und eine MGE-Grafikkarte.

Die MGE bietet die Möglichkeit, sogenannte "Hardware-Fenster" zu öffnen. Vier davon können gleichzeitig dargestellt werden. Da das Desktop bei der MGE ein Fenster belegt, bleiben noch drei Fenster für andere Rechner darzustellen. Wie arbeitet nun diese Funktion? Nach einer frei festlegbaren Zeit wird vom fernzusteuernden Computer der Bildschirm zum Server übertragen. Obwohl hier immer 32kB übertragen werden müssen, dauert die Übertragung nur den Bruchteil einer Sekunde - quasi nicht feststellbar. Der Server kann nun das Hardware-Fenster, in dem der Bildschirm des Servers dargestellt wird, frei auf seinem eigenen Monitor positionieren. Sobald man auf dem Server mit der Maus in ein Hardware-Fenster gelangt, besitzt man die volle Kontrolle über den Rechner, zu dem das übertragene Bild gehört. Soll heißen: Alle Mausbewegungen werden sofort vom Server zum entsprechenden Client übertragen und dort auch ausgeführt. Das hört sich etwas kompliziert an, deshalb ein Beispiel: Benutzer A (der Client) arbeitet gerade mit Script, Benutzer B (der Server) benutzt ein anderes Programm. Wenn nun Benutzer A auf ein Problem stößt und nicht mehr weiterkommt, fragt er Benutzer B. Schön und gut, aber was macht A, wenn B 3 Kilometer entfernt sitzt? Kein Problem, wenn man vernetzt ist. B öffnet einfach ein Hardware-Fenster und läßt per Bagdad den Bildschirm von A übertragen. Fährt er mit seiner Maus in das Fenster hinein, kann er den Rechner von A, also in diesem Fall Script, komplett (!) steuern! Mausbewegungen und -klicks werden ebenso ausgeführt wie Tastendrücke.

Wie gesagt, lassen sich drei Hardware-Fenster gleichzeitig öffnen. Entsprechend dieser Beschränkung können gleichzeitig drei STs ferngesteuert werden. Natürlich kann es dabei etwas eng auf dem Bildschirm werden. Aus diesem Grund lassen sich die Fenster frei auf dem Bildschirm positionieren. So können die Fenster an den Rand geschoben und dann hervorgeholt werden, wenn man sie benötigt.

Durch die Fernsteuerung ergeben sich natürlich vielfältige Möglichkeiten. Zum einen lassen sich in bereits bestehenden Netzwerken die Arbeiten der Mitarbeiter an einem zentralen Rechner überwachen: Der Server hat stets Kontrolle über seine "Zöglinge". Natürlich kann der Server so auch Hilfestellung geben, wenn einer der Clients Schwierigkeiten bei der Bedienung eines Programms hat oder bei einem anderen Problem nicht mehr weiterkommt. Eine weitere Möglichkeit besteht darin, daß eine Person bis zu vier Rechner gleichzeitig bedient. Während sie auf dem ersten ST einen Text schreibt, compiliert sie auf dem anderen ein Programm, auf dem dritten Rechner läßt sie sich Daten aus einer Datenbank zusammensuchen, vielleicht spielt sie auf dem vierten sogar noch ein Spiel...

Entfernungen

Interessant ist dabei, daß die Computer bis zu drei Kilometer entfernt voneinander stehen können. Dabei ist diese Entfernung garantiert, laut Hersteller hat es auch bereits mit einer Entfernung von 7 km funktioniert. Dadurch lassen sich auch Firmen mit verschiedenen Dienstgebäuden problemlos vernetzen, oder zwei Freunde, die nebeneinander wohnen, können ihre Rechner miteinander verbinden. Die Anwendungsgebiete von Bagdad sind fast nicht zu überblicken.

Natürlich lassen sich auch Zugriffsberechtigungen für die vernetzten Geräte vergeben, so daß man beispielsweise auf einen Bagdad-Rechner nur lesend zugreifen kann, auf einen anderen nur schreibend (zur Datenablage etc.). Bemerkenswert ist auch, daß trotz des erheblichen Übertragungsaufwands (ständig 32 kB pro Fenster) keine merkliche Verlangsamung eintritt. Der fernsteuerbare ST verhält sich genauso wie ein normaler ST.

Fazit

Bagdad ist das erste Programm seiner Art und überzeugt schon jetzt durch Komfort. Alle Einstellungen lassen sich per GEM-Dialog verändern. Während des Tests sind keinerlei Probleme aufgetreten; Bagdad ist die perfekte Möglichkeit, eine gleichzeitige Kontrolle über mehrere Rechner zu erhalten. Positiv zu bemerken ist auf jeden Fall, daß das Programm kostenlos mitgeliefert wird.

MP

Bezugsquellen:

Bagdad: PAM Software
Carl-Zuckmayer-Straße 27
6500 Mainz 33, Tel. 06131/476312

MGE: MAXON Computer GmbH
Schwalbacher Straße 52
6236 Eschborn, Tel. 06196/481811



IMAGIC präsentiert mit dem ST

Wer einmal auf einer der großen Computermessen gewesen ist, dem ist bestimmt auch eine Grafik-Show von IMAGIC GRAFIK aufgefallen. Doch auch zu anderen Gelegenheiten wie z.B. der letzten Internationalen Automobil-Austellung (IAA) konnte man die Präsentationsprodukte dieser jungen Firma bewundern, die auf dem ST den Begriff Desktop Video geprägt hat.

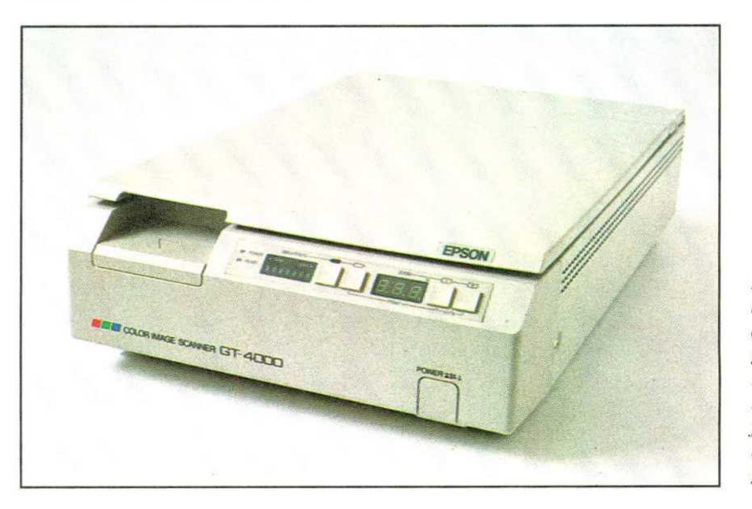
Begonnen hatte alles zu zweit im Jahre 1987. Damals entstanden erste Bilder-Shows für den ATARI ST. Bald schon genügten die normalen Möglichkeiten des Digitalisierens nicht mehr, so daß beschlossen wurde, eigene Software für Digitizer zu schreiben, um die Bilder in besserer Qualität in den ST zu bringen. Nachdem man nun über die entsprechende Software verfügte, wurden Überblendeffekte programmiert, damit sich die Bilder in den Shows nicht einfach nur aneinanderreihten. Daraus entwickelte sich dann mit der Zeit ein richtiger Showgenerator mit eigenem Zeichenteil und pascalähnlicher Programmiersprache. Der Bekanntheitsgrad wurde größer, als dieser Show-Generator seinen Weg zur Heidelberger Firma Application Systems fand und seitdem unter dem Namen IMAGIC vertrieben wird. Fortan konnte man auf jeder ST-Messe auf dem Stand von Application Systems neue Bilder-Shows bewundern, die sich immer wieder durch hervorragende Qualität und Originalität auszeichneten. So wurden z.B. mehrere STs über die MIDI-Schnittstelle verbunden und so eine ganze Monitorwand mit einer über alle Monitore fortlaufenden Show synchronisiert. (Es lassen sich mit IMAGIC übrigens bis zu 256 Rechner miteinander synchronisieren.) Durch schnelle Pack- und Entpack-Algorithmen konnte man sich auch im Bereich der Animation von tanzenden Mädchen und dem bekannten Animationsfilm "Aladin's Lampen" überzeugen, der dem Original "nachdigitalisiert" wurde.

Wer regelmäßig die ST Computer liest, konnte ja auch unsere Serie "Bildwerkstatt ST" verfolgen, die auf IMAGIC basierte und mit zwei PD-Disketten, auf denen ein Mini-IMAGIC enthalten ist, unterstützt wurde.

Im Business

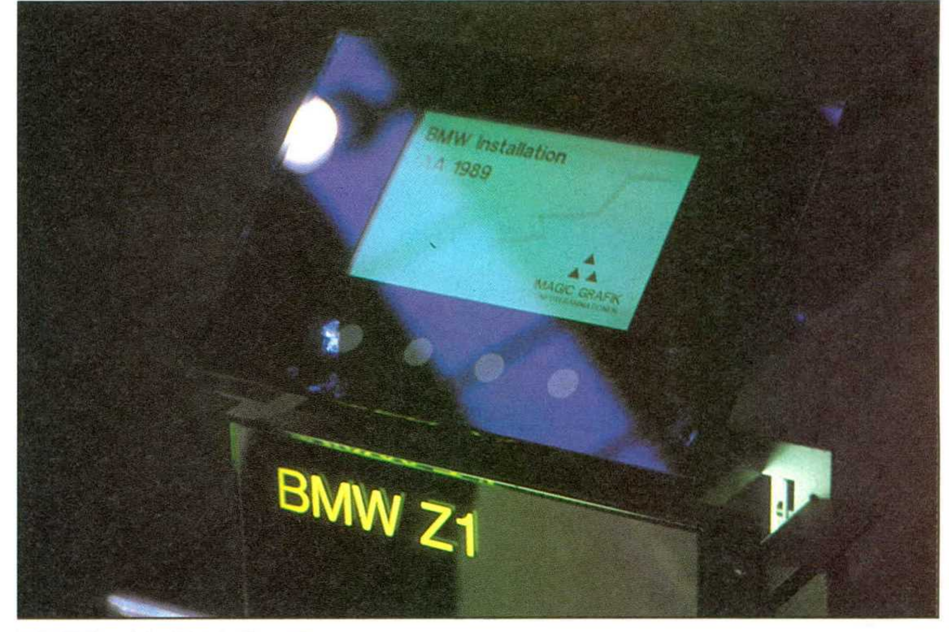
Im Laufe der Zeit kristallisierte sich aber immer mehr heraus, daß der ideale Markt für die IMAGIC GRAFIK-Produkte in der kommerziellen Werbung und im Business zu suchen ist. Unterstützt durch weitere freie Mitarbeiter aus den Bereichen Informatik und Elektrotechnik hat man sich mittlerweile auf den professionellen Markt gestürzt. Die Anwendungen reichen von Echtzeitanimationen bis zu Messepräsentationen. Hierbei wird in der Zwischenzeit auf Wunsch auch jede Menge Elektronik benutzt. So sind z.B. Synchronsteuerungen für Bildplattenspieler und U-Matic-Player problemlos möglich. Für Messeeinsätze etc. werden die verwendeten Geräte zuvor mit einigen Feinheiten modifiziert, denn Messezeit ist nicht nur für das Personal Stresszeit, auch Computer werden auf das höchste beansprucht können.

Auf der letzten Internationalen Automobil-Ausstellung in Frankfurt konnte man auf dem BMW-Stand vor jedem



Eines der vielen technischen Geräte, die optimale Bilder garantieren, ist der Farbscanner EPSON GT-4000, für den eine eigene Scan-Software geschrieben wurde.

Fahrzeug ein interaktives Dialogsystem finden, d.h. eine Konsole, die einen Mega ST beherbergte und ein hochauflösendes LC-Display ansteuerte. Auf der Konsole konnte man mittels Touchscreen dann die einzelnen technischen Daten zum jeweiligen Fahrzeug abrufen. Dieses Präsentationsverfahren sorgte sowohl bei den Mitarbeitern von BMW als auch bei den Besuchern der IAA durchaus für einiges Aufsehen.



IMAGIC auf der IAA in Frankfurt

The Times Are A-Changing

Natürlich hat sich mittlerweile auch das Handwerkszeug geändert. Die einfache Videokamera mußte einem Profigerät für knapp 20.000 DM weichen und für viele Anwendungen bieten Farbscanner, Grafikkarte und Genlock-Technik die ideale Ergänzung zur Kamera.

IMAGIC GRAFIK bietet sogar ein eigenes Genlock-Interface für semi- und professionellen Einsatz auf dem ST an. Mit diesem kann man beliebig Computer- und Videobilder miteinander mischen und auf einem Bildschirm darstellen. Ideal ist hier z.B. der Einsatz als Titelgenerator. Aber auch weitere Bausteine für Videoanwendungen wie Encoder- und Decoder-Systeme für die Aufzeichnungen auf Video sind bei IMAGIC GRAFIK erhältlich.

Wichtig für die optimale Bildqualität, die fast schon an Fotos herankommt, ist auf jeden Fall nach wie vor die Software, die im Hause IMAGIC GRAFIK selbst entwickelt wird. Für Texteinblendungen wurden eigene Zeichensätze kreiert, die bei Produktpräsentationen vielfach zum Einsatz kommen. Es lassen sich allerdings auch alle Signum!-Zeichensätze für Demos verwenden, die mittels Zusatz-Software entsprechend angezeigt werden.

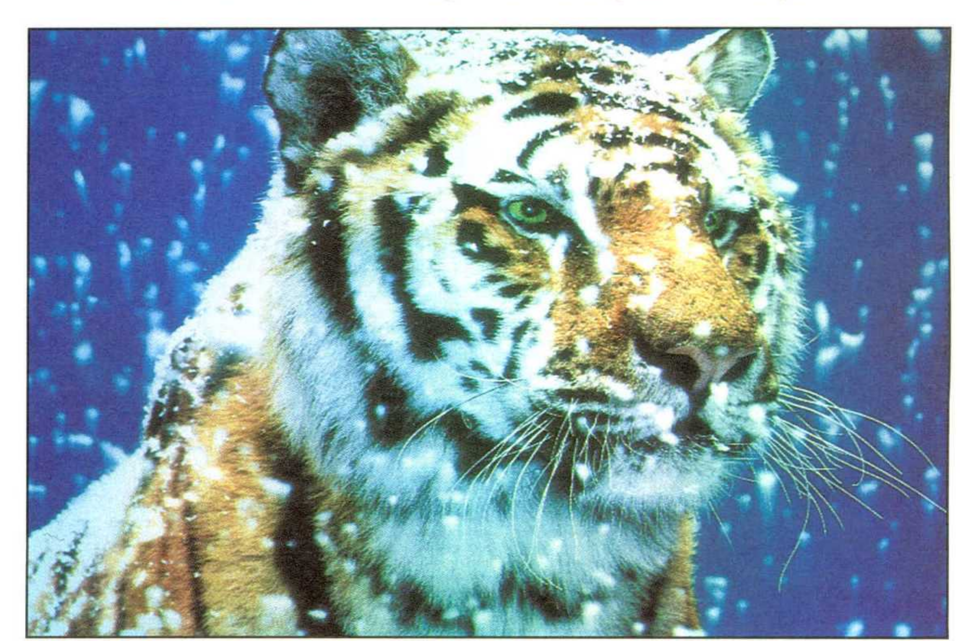
Zur Zeit entsteht eine Grafik-Show auf einer MAXON-Grafikkarte MGE, die dann auf der ATARI-Messe in Düsseldorf am MAXON-Stand und bei ausgewählten Händlern zu bewundern sein wird. Einen kleinen Vorgeschmack geben die zwei der nebenstehenden Bilder.

Zum Einsatz kommen wird dabei die neueste Version der MGE, die nun 256 aus 16,7 Mio. Farben oder echte 256 Graustufen unterstützt, und ein 95 cm-Großbildschirm. Man darf also gespannt sein!

HE



Diese beiden Bilder wurden mit 256 Farben auf der MAXON-Grafikkarte MGE dargestellt.



Kontaktadresse:

*IMAGIC GRAFIK
Computeranimationen
Waldenbucher Str. 53
7447 Aichtal-Aich
Tel.: 07127/57829*



5000 Köln 41 Sülz Mommsenstr. 72

Ihr Fachhändler in Köln für Atari / XT / AT Tel.: 02 21/43 01 442, Fax 46 65 15
Wir bieten Ihnen noch Beratung und Service für Ihren Computer

SCSI-Festplatten, die schnellen 580 KB/s, 1 Jahr Garantie
20 MB 40 ms anschlußfertig mit Software, von Protar 798,-
30 MB 40 ms Vortex Datajet mit Software 1298,-
40 MB 40 ms SCSI Platte, 1 Jahr Garantie von Protar 1299,-
40 MB 19 ms SCSI Quantum, 2 Jahr Garantie von Protar 1498,-
80 MB 19 ms SCSI Quantum, 2 Jahre Garantie von Protar ... 1998,-
105 MB 19 ms SCSI Quantum, 2 Jahre Garantie 2098,-

ST Mega 1/SM 124 mit 1 MB 1598,-
ST Mega 1/SM 124 mit 4 MB 2698,-
Desktop-Anlage ST 4 MB / 30 MB / Laser / Calamus 6998,-
ST Mega1/SM 124 mit 2,5MB 2048,-
Scanner Panasonic 400x400 RS 505 1999,-

PC Speed mit Einbau in ST 24 Std. 498,-
ST. Laufwerk o. Bus 3.5 Zoll NEC Lw. 229,-
ST Laufwerk40/80 5.25 Zoll TEAC 298,-
ST Laufwerk 40/80 5.25 Zoll Bus TEAC 348,-
NEC Lw roh für Einbau in ST 1040 190,-
ST Laufwerk roh 3.5 TEAC 1.44 MB 230,-
NEU AT EMULATOR Vortex 80286 für ST
zum Einbau in ST m. EGA-Emul., sof. lieferb. **498,-**

Speicherkarten 2.5 MB o. Ram für ST 149,-
512 KB Karte bestückt steckbar 249,-
4 MB Karte mit 2 MB bestückt 688,-
Speicherkarten 4 MB 239,-
2.5 MB Speichererweiterung 598,-
4 MB Speicherkarte 998,-
Einbau der Speicherkarten in Ihren ST 98,-

Eizo Monitor 9060 S 1598,-
Multisync S/W 548,-
Monitorkabel 69,-
Switchbox 2 Mon an ST m. Softw. 45,-
HF-Modulator 198,-
ST-Tastatur Geh. 140,-
Adimens 3.0 398,-
Mega Paint 498,-
Mega Paint Prof. 798,-
Signum!2 Text 388,-
Tempus 2.06 119,-
STAD Grafik 1.3 149,-
Script Text 159,-
Freesoftw. aus ST10 St. nur 50,-
Freesoftware einzeln 6,-
über 800 PD Disk Info anford. gegen 5,-
Mega Paint 2 450,-
Calamus 698,-
Calamus Font n. Wahl
Fax Pock. Modem 2400/4800 498,-
Fax Software nur für PC
Modem Discovery 2400/1200/300 348,-
Spectre 128 o. Rom's 398,-
Die Inbetriebnahme der Modeme am öffentlichen Postnetz der BRD

NEU: NEC PT60 Drucker 24 Nadeln 300 Z/Sek. A4 leise . 1598,-
Citizen Swift 24 Nadeln, 2 Jahre Garantie 948,-
HP Deskjet 1698,-
Panasonic 1124 24 Nadeln 848,-
Atari Laser SLM 804 2650,-

Atari/Star/Schneider/Panasonic sind eingetragene Warenzeichen. Wir liefern für Ihre Firma die richtige Soft/Hardware/Beratung und Aufstellung. Faktura für XT/AT PC Komplettsystem mit Einweisung Info im Laden. Öffnungszeiten 10.00-13.00 Uhr, 14.00-18.00 Uhr Samst. 10.00-14.00 Uhr

Hendrik Haase Computersysteme präsentiert:

Atari-Computer

Atari 1040 STE incl. SM 124 — 1448,- DM

Atari Mega ST - Computer in unterschiedlichen Versionen zu absoluten TOP-Konditionen

Stacy und TT - Computer

Atari Megafile 30 Festplatte	880,- DM
Atari Megafile 60 Festplatte	1240,- DM
Vortex Datajet 30 Festplatte	1180,- DM
Vortex Datajet 60 Festplatte	1680,- DM
Star Drucker LC 24/10	668,- DM
Atari Laserdrucker SLM 804	2600,- DM
Mitsubishi Multiscan	1298,- DM
NEC 3D Multisync	1550,- DM
Vortex AT Emulator (286 !!)	498,- DM

Andere Produkte zu TOP-Konditionen auf Anfrage

Bestellungen und Informationen bei:

Hendrik Haase Computersysteme
Wiedfeldtstraße 77 • D-4300 Essen 1
Telefon 02 01 - 42 25 75 • Fax 02 01 - 41 04 21



Das Modul-Lexikon für den ATARI ST

'Lex-o-Thek': das Grundprogramm — *DM 49,-*
'Lex-o Thek' ist ein Accessory, mit dem mehrere Module bedient werden können. Das Programm besitzt eine Schnittstelle zu den Textverarbeitungen '1st Word Plus' und 'That's Write'. Alle Begriffe können direkt aus dem Lex-o-Thek-Fenster in Ihre Texte übernommen werden. Das Programm ist voll mit der Maus oder der Tastatur bedienbar. Das Grundprogramm ist zum Betrieb der Module erforderlich.

Modul 1: '3rd Word', das Synonymenlexikon — *DM 89,-*
'3rd Word' enthält 65 000 Begriffe. Nach Eingabe eines Suchbegriffs in das 'Lex-o-Thek'-Fenster werden ganze Blöcke sinnverwandter Begriffe angezeigt. Nach jedem angezeigten Begriff kann weiter verzweigt werden. So findet jeder das treffende Wort und wird beneidet für seine excellenten Formulierungen.

Modul 2: 'Herz-Schmerz', das Reimlexikon — *DM 69,-*
'Herz-Schmerz' ist nicht nur für Dichter und Denker interessant. Mit diesem Modul bekommen Sie zu jedem eingegebenen Wort hunderte Reimwörter angezeigt. Ein ausgeklügeltes System und die Verzweigungsmöglichkeit ermöglichen es auch dem Hobbydichter, sich zu jedem Anlaß einen passenden Reim zu machen.

Modul 3: 'Bonmot', die Zitatedatenbank — *DM 69,-*
'Bonmot' enthält eine Vielzahl klassischer und moderner Zitate, Sprichwörter, Bonmots, Spontisprüche, Definitionen, Bibelsprüche und Bauernregeln. Bei der Auswahl wurde großer Wert auf die Alltagstauglichkeit der Zitate gelegt. Die Zitate werden durch ausführliche Sachregister, Stichwortregister und Autorenregister erschlossen. So findet jeder, der Texte oder Reden schreibt, sofort das richtige Zitat.

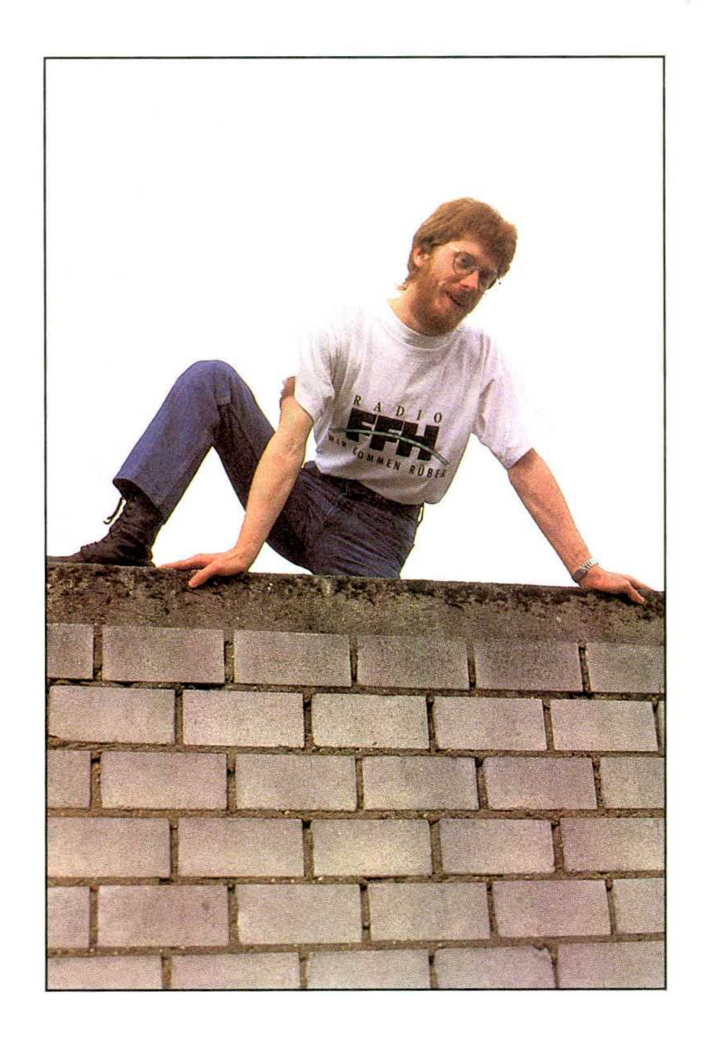
Komplettpaket-Preis (Ersparnis: 27,- DM) — *DM 249,-*
Preise zuzügl. Versandkosten: Vorkasse 5,- DM, NN 8,- DM



Reinhard Rückemann
Grundstrasse 63
5600 Wuppertal 22
02 02 / 64 03 89

RADIO FFH - Wir kommen rüber

Radiosender mit ST



Der Strom der professionellen Anwendungen mit dem ST reißt nicht ab. In Hessen arbeitet einer der größten privaten Radiosender Deutschlands mit einem ATARI ST. Jingles, Spots, Bumper, Beds und Stinger mit dem ST - RADIO FFH macht's möglich!

100,2 MHz - inzwischen ist diese Frequenz diejenige, die an meinem Radio fest eingestellt ist. Auf dieser Frequenz sendet der hessiche Sender RADIO FFH rund um die Uhr sein Programm. Wenn es um punkt 16 Uhr wieder "RADIO FFH - Feierabend" heißt, sollten alle hessischen ST-Freaks das Radio aufdrehen, denn hier steuert ein Notator von C-Lab den sogenannten "Bumper".

Von 8 auf 100,2 in 0 Sekunden

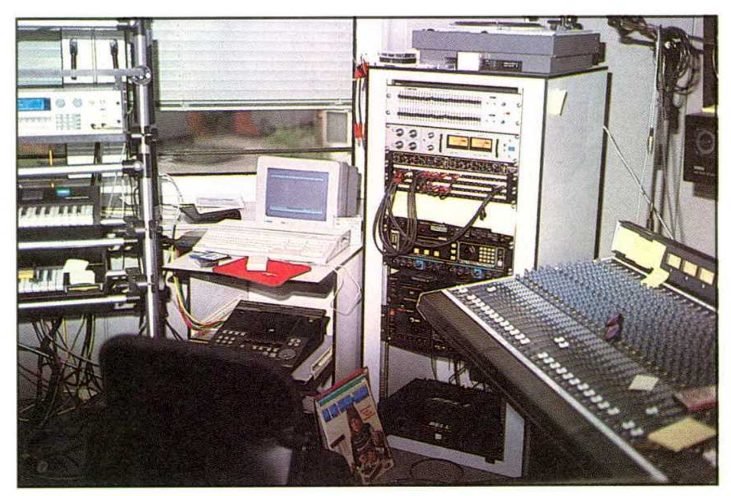
8 MHz ist die Taktfrequenz des ST - 100,2 eine der FFH-Frequenzen. Ingo Gsedl, ein Musiker und Produzent aus Berlin, hat bei FFH seinen Arbeitsplatz gefunden. Inmitten von 16-Spur-Tonanlagen, Studios, Tonbändern, Sequencern und Keyboards arbeitet er mit dem ST neue Jingles aus, die das Radioprogramm unterstützen, verfeinern und interessanter machen sollen. Ein Großteil der Jingles stammt von "Killer Music" aus Los Angeles und wird von Ingo Gsedl in zahlreiche verschiedenste Variationen verwandelt (s. u.). Die Arbeit bei einem Radiosender ist allerdings meist sehr spontan, manchmal muß ein Spot innerhalb weniger Stunden oder gar Minuten produziert werden - in diesem Fall kann man natürlich nicht "mal eben" in Los Angeles produzieren lassen, sondern hier ist Zeit für Ingo Gsedls Auftritt. Mit einem ST, einem Unitor (einer erweiterten Schnittstelle zum Lesen und Schreiben des SMPTE-/EBU-Time-Codes mit zusätzlichen MIDI-Ein- und Ausgängen), einem Akai S1000, zwei Keyboards und vielem anderem Radiosender-Zubehör produziert er in Windeseile kurze Musikstücke, die dann unter der Sprecherstimme abgespielt werden.

Ein Stück besteht dabei aus einem Bumper, einem Bed und einem Stinger, andere nur aus einem Bumper und einem Stinger. Bumper nennt sich der Anfang eines Beitrags. Wenn etwa der Sprecher "RADIO FFH - Wir küssen die Behörden wach" oder "5 vor 4 - RADIO FFH - Nachrichten - ich bin Maren Mende, guten Tag" sagt, wird das mit einem Bumper gestartet. Nehmen wir zum Beispiel den RADIO FFH-Feierabend (montags bis freitags von 16 bis 18 Uhr): Wenn die Moderatorin Eva Mayer zwischen den Liedern einen vorproduzierten Beitrag abfährt, kann es oft vorkommen, daß eben dieser mit einem Bed unterlegt ist. Kommt es zum Schluß des Beitrags, wird es interessant: Sobald der Beitrag beendet ist, geht das Bed über in den Stinger - ein Stinger ist das Musikstück am Schluß eines Beitrags. Interessant dabei ist, daß Bumper, Bed und Stinger bei geschickter Überleitung harmonisch ineinander übergehen und man gar nicht merkt, daß es sich hier um verschiedene Stücke handelt, die jeweils von einer eigenen Bandmaschine gestartet wurden. Das vorproduzierte Bed ist dabei länger als der längste zu erwartende Wortbeitrag, damit auf jeden Fall jede Stelle musikalisch unterlegt ist.

Aus eins mach zwei

Programmelemente werden immer dann eintönig, wenn man sie zu oft hört. Aus diesem Grund müssen viele verschiedene Jingles existieren, die zwar alle auf dem gleichen musikalischen Thema basieren, sich aber verschieden aufbauen. Die Zusammensetzung solcher Jingles, die zwar unterschiedlich sind, sich aber auf eine bestimmte Weise doch gleichen, fällt in der Zusammenarbeit des STs in Verbindung mit dem Sampler leicht. Die Killer Music-Stücke aus Los Angeles werden beispielsweise in Einzelteile aufgespalten und neu zusammengesetzt. Hier und da kann natürlich auch ein Solo-Instrument hinzugefügt werden, um neue Klangeindrücke zu erzielen. Auf diese Art und Weise entstehen immer wieder neue Variationen eines Musikstücks und den FFH-Hörern wird es nicht zu eintönig. Der Vorteil liegt auf der Hand: Trotz verschiedener Stücke bleibt der Wiedererkennungswert erhalten und jeder Hörer weiß sofort, welchen Sender er gerade eingeschaltet hat.

Der ST ist natürlich auch dazu zu benutzen, via MIDI-Schnittstelle und Time-Code-Synchronisation zu den vorhandenen analogen Tonbandanlagen beliebig viele Synthesizer und Sampler parallel zu betreiben und diese gemeinsam mit den 16 Analogspuren der Bandmaschine über das Mischpult abzumischen. In diesem Fall stellt lediglich die Anzahl der Mischpultkanäle die Begrenzung der möglichen imaginären "Spuren" dar. Beim späteren Abspielen des fertigen Bands fällt dann nicht mehr auf, daß sich hier eigentlich viel mehr Spuren tummeln - vom ST gesteuert. Bei der direkten Abmischung der Synthesizer und Sampler - ohne sie auf die analoge Mehrspurmaschine aufzunehmen - steigt sogar die Tonqualität.



Bei Radio FFH findet man den ST zwischen teuren Steuerpulten.

Es hallt zurück

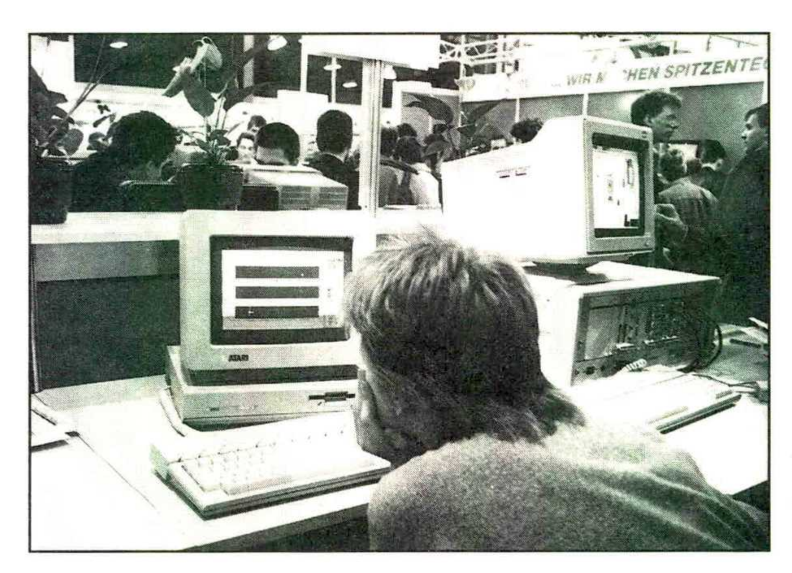
Damit ist die Reihe der möglichen ST-Anwendungen nicht beendet. Auch Hallgeräte wie z.B. das "Lexicon LXP-1" werden mit dem ST angesteuert und ediert. Dadurch ergeben sich Möglichkeiten, die mit den Bedienelementen des Hallgeräts alleine nicht möglich wären. So lassen sich bei einem Effekt beispielsweise die Hallzeit, der Frequenzverlauf oder das Feedback (also die Anzahl der vorhandenen Echos) ohne große Schwierigkeiten verändern. Für diesen Zweck hat sich Ingo Gsedl sogar eigens ein Programm geschrieben, das die Aufgabe für ihn erledigt.

Die Frage, warum ein so großer Radiosender wie RADIO FFH sich mit einem ST herumschlägt, ist recht einfach zu beantworten: Die Preise für digitale Arbeitsplätze wie das Syn-Clavier oder die Waveframe-Workstation, an denen allein digital aufgenommen, geschnitten und abgemischt werden kann, liegen im sechsstelligen Bereich. Natürlich ließe sich damit die Arbeit rationeller und auch schneller bewältigen, hier aber reicht der ST (noch) aus - der findige Gsedl ist mit dem Gerät mehr als zufrieden und möchte es auf keinen Fall mehr missen - besonders die MIDI-Schnittstelle und die zahlreich vorhandenen Musikprogramme haben es ihm angetan.

Wo Sie RADIO FFH hören können? Leider nur, wenn Sie in Hessen wohnen. In diesem Fall allerdings müssen Sie einfach nur Ihr Radio auf UKW einschalten und eine der folgenden Frequenzen wählen: Kassel 103,7; Eschwege 104,6; Dillenburg 100,0; Rimberg 100,3; Driedorf 106,8; Vogelsberg 104,8; Rhön 100,9; Frankfurt 100,2; Wiesbaden 102,0; Dieburg 90,1 und Südhessen 105,0. Inklusive Nachrichten, Musik, Sport, Verkehrsfunk mit ARI-Piepser - und einem ATARI ST.

MP

Let's Go East



ATARI auf der Leipziger Messe - ein begehrter Rechner

ATARI-Club in der DDR

Im Februar dieses Jahres gründete der Dresdner Thomas Wedler einen ATARI-Club. Dieser Club soll zu einem Dachverband ausgebaut und über ihn der Vertrieb von ATARI-Computern und Zubehör in die DDR abgewickelt werden.

Die DDR war schon vor der großen Wende 1989 ein "ATARI-Land". Mehr als 100000 ATARIs sind allein über die Handelsorganisation Forum in den Intershops verkauft worden, schätzte ATARI auf der Leipziger Messe im März dieses Jahres, Im wesentlichen Low-End-Geräte, also 8-Bit-Rechner, 520 STs, 1040 STs und die XTs. Eine unbekannte Zahl von Geräten brachten Omas von ihren Westreisen mit oder schickten die Verwandten von "drüben". Mehrere Universitäten, darunter die Karl-Marx-Universität in Leipzig und die Humboldt-Universität in Berlin, setzten in größeren Mengen ATARIs ein, und zu den ersten freien Volkskammerwahlen am 18. März stellte ATARI für Wahlauswertung und Politbarometer 80 Computer zur Verfügung. Auch komplette Zeitungen werden inzwischen mit ATARIs gesetzt, so zum Beispiel "Die Andere Zeitung" (DAZ) des neuen Forums in Leipzig.

Was lag also näher, als einen ATARI-Club zu gründen? Tatsächlich hatte Thomas Wedler schon 1988 diese Idee, aber damals paßte das nicht in die politische Landschaft. Als sich dann die ersten Veränderungen im Oktober 1989 anbahnten, nutzte der Dresdner die Gelegenheit, um sein "Projekt ATARI" in die Tat umzusetzen. "Ich fand bald einen kleinen Kreis von 15 bis 20 Usern, die von meiner Idee begeistert waren", erzählt er. "Rechtliche und materielle Unterstützung fand ich beim Kulturbund der DDR. Bedingung war allerdings, daß alle Club-Interessenten bereit waren, beim Kulturbund mitzumachen, um zu dessen Finanzierung beizutragen. Seit der Streichung der Gelder von der damaligen SED ist das sogar zum Dreh- und Angelpunkt für die Bezahlung der hauptamtlichen Mitarbeiter beim Kulturbund geworden.", beschreibt Wedler die komplizierte Situation.

Rund 40 Mitglieder im Alter von 14 bis 60 Jahren kamen schließlich im Dezember 1989 zusammen, als ihrem Club offiziell erlaubt wurde, sich "IG ATARI - Computerclub im Kulturbund der DDR" zu nennen. Eine Veröffentlichung in der Dresdner Bezirkspresse im Januar sorgte für eine "Flut von Briefen von Interessenten", erzählt Wedler, "aber zum Glück hatte ich mein Postamt und meinen Briefträger vorher informiert". Im Februar fand dann die offizielle Gründungsversammlung statt, und der Club gab sich eine ordentliche Satzung. Thomas Wedler wurde ehrenamtlicher Geschäftsführer.

"Hauptziel ist die Schaffung eines Kommunikationszentrums für alle ATARI-Freaks der 8- und 16-Bit-Technik." sagt der unternehmungslustige Dresdner. Über 150 Mitglieder hat die IG derzeit, und zusammen mit den Leitern der Arbeitsgruppen und dem "Schatzmeister", die in der Übergangsphase die Geschäfte des Clubs leiten, baut Wedler ein Netz von Kontaktstellen im gesamten Bezirk Dresden und in Leipzig auf. Sie bemühen sich um die Bildung von Ortsgruppen ab 5

Usern, die selbständig vor Ort arbeiten. Aufgenommen werden einzelne User und bereits bestehende Clubs. Eine eigene Clubzeitung soll herausgegeben und ein Landesverband Sachsen gegründet werden. Auch eine feste Anlaufstelle gibt es in Dresden bereits.

Die IG ATARI bietet Mitgliedern und Interessierten vielfältige Arbeitsmöglichkeiten. Neben individuellen Zusammenkünften stehen den Mitgliedern verschiedene Arbeitsgruppen offen. So steht bei der AG Software "Programmieren in ATARI-BASIC, Assembler und anderen ATARI-Sprachen" auf dem Plan. In der AG Trickkiste werden Fragen behandelt, die beim Arbeiten mit Anwender- oder Spielprogrammen auftreten, Tips und Tricks vermittelt und außerdem Beschreibungen zu bestimmten gefragten Programmen erstellt. Die AG Sound hilft beim Einsatz des ATARI als Sound-Maschine.

Ein DDR-spezifisches Problem wird in der AG Hardware behandelt: Die Hardware-Reparatur. "Hier in Dresden gibt es zwar einen Computerservice", erklärt der Club-Chef, "aber die Reparatur muß in DM bezahlt werden. Darum bleibt meist nichts anderes übrig, als es mit Basteln zu versuchen. In unserer Hardware-Gruppe haben wir dafür ausgesprochene Spezialisten, aber wenn neue Bauteile gebraucht werden, muß man immer noch in den Westen fahren und gegen DM kaufen oder man läßt es ganz."

Die AG Organisation ist für die Öffentlichkeitsarbeit und Organisation verantwortlich. Sie stellt die Kontakte des Clubs nach außen her, hält die Verbindung mit dem Kulturbund aufrecht und gibt eine Übersicht über clubeigene Software und Literatur. Sie organisiert außerdem Mitgliederversammlungen und öffentliche Treffs. Zu den weiteren Vorhaben gehören eine AG Sprachsynthese, AG Grafik/Design, AG LOGO und eine AG Club-Magazin.



Über 100000 ATARI-Rechner wurden im Intershop verkauft.

Ein großes Mitgliedertreffen in Form einer Börse wird alle Vierteljahre veranstaltet. Das erste dieser Art in Dresden überhaupt fand im März diesen Jahres statt. "Dieser Art" bedeutet öffentlich, denn bisher trafen sich die ATARI-Freaks in Schulclubs oder in größeren Wohnungen. Auf der Börse konnten Software, Erfahrungen und Informationen ausgetauscht werden, und in Zusammenarbeit mit bundesdeutschen Firmen wurde erstes ATARI-Zubehör angeboten, zum Beispiel ATARI-Lichtgriffel und vieles andere. Eine Marktanalyse sollte den Bedarf an Diskettenstationen und Druckern klären. Der rührige Wedler will außerdem ein ATARI-Verkaufsbüro in Dresden eröffnen, "wo Computerfreunde, IG-Mitglieder natürlich bevorzugt, sowohl Computer als auch Zubehör erwerben können. Früher konnten sich viele einen Computer nicht leisten, weil eine Floppy oder ein Drucker made in DDR für einen Wucherpreis von 5000 Mark verkauft wurden und weil DM knapp war, darum gibt es hier jetzt einen enormen Markt. Zentrale Anlaufstelle für Vertrieb und Verkauf soll die IG ATARI sein. Entsprechende Kontakte mit einigen ATARI-Fachgeschäften und Software-Häusern in der Bundesrepublik sind bereits geknüpft."

Nicht kommerzielle Kontakte sucht die IG zu ATARI-Clubs in der Bundesrepublik, "vor allem zum Erfahrungsaustausch, zum Beispiel über Raubkopien, jugendgefährdende Software und überhaupt alles, was die Arbeit mit ATARIs betrifft", meint Wedler.

Im Club selbst geht es streng demokratisch zu: Im Mai findet die nächste Mitgliederversammlung statt, auf der die Clubleitung, die zunächst einmal alle drei Monate Zeit hatte, sich einzuarbeiten, für die kommenden zwei Jahre gewählt wird. Ihre Aufgabe ist vor allem, die reibungslose Arbeit der einzelnen Mitglieder zu berücksichtigen. In der Clubordnung wird dabei auch an einen verantwortungsbewußten Einsatz der Gelder appelliert, denn der Club finanziert sich selbst. Die ebenfalls für zwei Jahre zu wählende Revisionskommission wacht darüber.

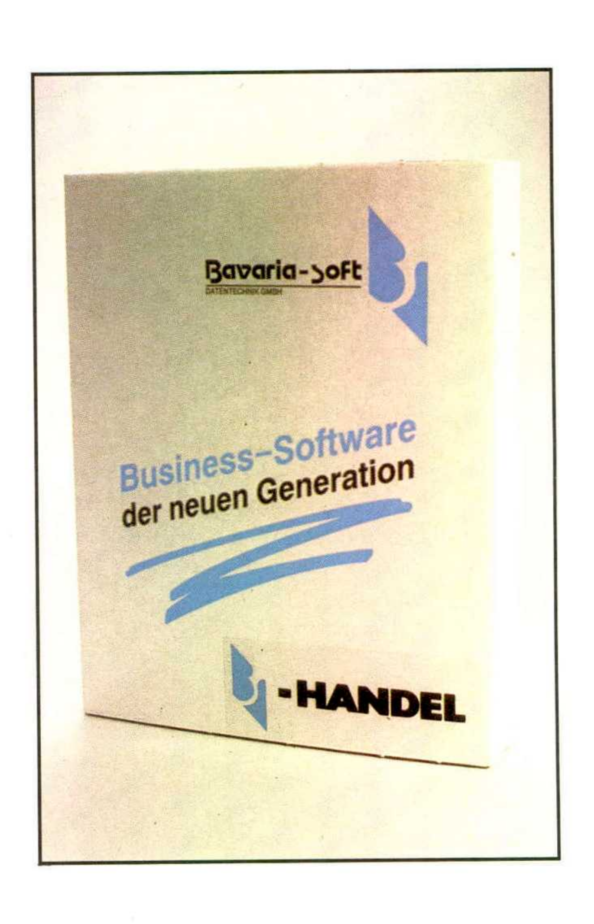
Interessenten können sich direkt an den Computer-Club wenden:

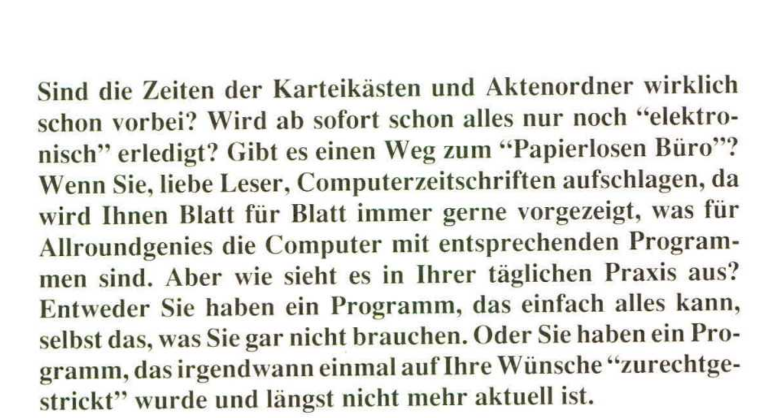
IG ATARI
Alexander-Herzen-Straße 55
DDR - 8080 Dresden

Andrea Rudolph

Kaufmann inklusive

BS-Handel/3





Sind die Zeiten der Karteikästen und Aktenordner wirklich schon vorbei? Wird ab sofort schon alles nur noch "elektronisch" erledigt? Gibt es einen Weg zum "Papierlosen Büro"? Wenn Sie, liebe Leser, Computerzeitschriften aufschlagen, da wird Ihnen Blatt für Blatt immer gerne vorgezeigt, was für Allroundgenies die Computer mit entsprechenden Programmen sind. Aber wie sieht es in Ihrer täglichen Praxis aus? Entweder Sie haben ein Programm, das einfach alles kann, selbst das, was Sie gar nicht brauchen. Oder Sie haben ein Programm, das irgendwann einmal auf Ihre Wünsche "zurechtgestrickt" wurde und längst nicht mehr aktuell ist.

Alles nur Extrembeispiele - meinen Sie? Seien Sie einmal ehrlich, wie ausgelastet ist Ihr Computer denn wirklich? Gerade das Abwägen auf das Machbare ist eine wichtige Voraussetzung, um vor allen Dingen die richtige Software zu finden. (Und der Autor dieser Zeilen weiß wovon er schreibt, schließlich war er drei Jahre lang im Software-Verkauf tätig.) Kann es sich ein Klein- oder mittelständischer Betrieb erlauben, aufs Geratewohl Software auszuwählen? Sicherlich nicht. Gefragt sind Programme, die sowohl branchentypische Tätigkeiten abdecken als auch Offenheit in andere Richtungen bieten. (Man erinnere sich an obige Extrembeispiele.)

BS-Handel ist für die ATARI-ST Computer bereits ein Software-Klassiker. Seit 1986 auf dem Markt gehört BS-Handel im Bereich integrierte kaufmännische Software sicher zu den ersten Programmen dieser Art auf dem ATARI ST. Jetzt ist die neue Version "BS-Handel/3" lieferbar. BS-Handel/3 ist keine Update-Version, sondern stellt ein komplett neu entwickeltes Programm dar. Es beinhaltet nicht nur den Funktionsumfang der Vorgängerversion, sondern wurde in der GEM-Oberfläche und bei der Verarbeitungsgeschwindigkeit völlig neu überarbeitet.

Vor den Erfolg ...

Um es gleich vorwegzunehmen, ohne einen Hauptspeicher mit mindestens 1 MByte RAM und einer Festplatte mit 20 MByte sollten Sie erst gar nicht beginnen - BS-Handel/3 ist mit weniger leider nicht zufrieden! Bei einer Betriebssystemversion vor dem "Rainbow" (TOS 1.4) muß unbedingt auch das Prorämmlein "FOLDR200.PRG" geladen werden, da es zu dem besagten "40-Ordner-Problem" kommt. Auch ein Textverarbeitungsprogramm sollte verfügbar sein. Im Handbuch liest man: "Das Textprogramm ist ein wichtiger Bestandteil im BS-Handel/3." Das soll aber nicht heißen, daß ein solches mitgeliefert wird, als "wichtiger Bestandteil" im Lieferumfang. Im Gegenteil: Es wird nicht mitgeliefert und ist dennoch "wichtiger Bestandteil" bei der Arbeit, denn ohne Textprogramm können keine Ausdrucke gemacht werden! BS-Handel/3 bedient sich sehr gerne 1st_Wordplus, weil der Listengenerator darauf abgestimmt ist.

Vor den Erfolg haben die Götter bekanntlich den Schweiß gesetzt. So muß man bei BS-Handel/3 eine umfangreiche Installationsprozedur durchlaufen. Allein das Verteilen der einzelnen Dateien auf meine MEGAFILE-60 dauerte etwas mehr als 5 (in Worten: fünf) Minuten - da wird wohl der Plattentyp auch etwas schuld gewesen sein. Wenn es sich um eine sogenannte Erstinstallation handelt, erscheinen nach dem Programmaufruf verschiedene Masken für Grundeinstellungen: Wo werden Daten, Musterformulare, Druckertreiber in welchen Partitionen in welchen Pfaden zu finden sein? Feldbeschreibungen werden eingestellt, Anreden festgelegt, Versandart, Rabattsatz, Zahlungsbedingung und...und... und. Es dürfte aber dem Erstanwender plausibel genug sein, daß solche Einstellungen beim ersten Schritt in das Programm durchaus nötig und unumgänglich sind.

Einblicke, Ausblicke, Überblicke

BS-Handel/3 bietet uns im Einstiegsbild eine Desktop-Leiste mit nur vier Menüs und eine Funktionstastenleiste am unteren Bildrand. Dies ist erfreulich, denn die häufigsten Aufrufe erreicht man per Funktionstaste, selbst die weniger wichtigen Unterpunkte der Pull-Down-Menüs sind per Alt-Sequenzen zusätzlich ansteuerbar. Klar strukturiert sind die Menüs selbst: Stammdatenverwaltung, Auftragssteuerung, Ausgabevorbereitung und div. Kleinarbeit (hier Menü "Anweisung". Alles ist logisch zusammengefaßt. Interessant ist ein Druck auf Funktionstaste >F1<. Da rollt sich ein Pop-Up-Menü genau dort auf, wo gerade der Mauszeiger steht. Verbesserungsvorschlag von mir: Warum nicht eine Zweitbelegung der Funktionstasten vorsehen? Denn erst >F1< und dann Abkehr zur Maus und in Menüs ist nicht sehr ergonomisch.

Die Kartei bringt's

Ordnungsbegriff Nummer 1 ist die "Kartei". Sie beinhaltet gewissermaßen alle Stammdaten wie "Adressen", "Lager", "Preise" usw. Die nächste Ordnungseinheit ist der "Auftrag", der alle Tätigkeiten im Hinblick auf Kunden und Lieferanten bereithält. Punkt 3 sind die umfangreichen Ausgabemöglichkeiten an ein externes Textverarbeitungsprogramm.

Wenngleich wir bei der Installation die Globaldaten eingegeben hatten, müssen natürlich jetzt Adressen von Kunden und Lieferanten, Artikel, Preise und Anfangsbestände des Lagers erfaßt sein. Mit einem Druck auf >F1< und der Auswahl "Adressen" des besagten Pop-Up-Menüs wird nunmehr die maßgebende Kartei ausgewählt. Ab sofort wirken die Angaben im Menü "Stammdaten" nur auf diese eine Kartei. Also hinein in den Menüpunkt "Erfassen".

Eine ausführliche Maske erscheint, und man wird nach Name, Straße, Wohnort und alledem gefragt. Von hier aus kann in drei weitere Masken verzweigt werden, die zusätzliche Angaben zur 'Adresse' (z.B. Branche, Kreditlimit, Individualrabatt usw.), zur Umsatzstatistik dieser 'Adresse' und einen Notizblock aufnehmen. Nun muß nur noch festgelegt werden, ob es sich um einen Kunden oder einen Lieferanten handelt, denn diese zwei Adressentypen werden getrennt voneinander abgespeichert. In ähnlicher Manier werden Artikel, Bestandsdaten, Preise, Rabatte und Zahlungsbedingungen aufgenommen.

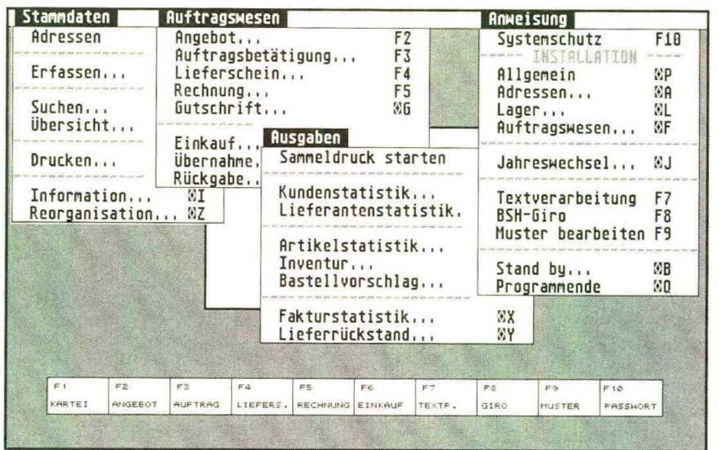


Bild 1: Eine überschaubare Ansammlung von Pull-Down-Menüs wird zur Programmsteuerung unterstützt von den zusätzlich aktivierten Funktionstasten.

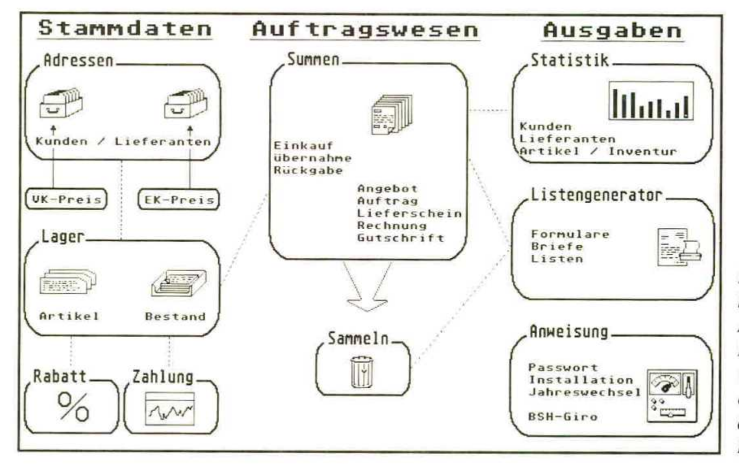


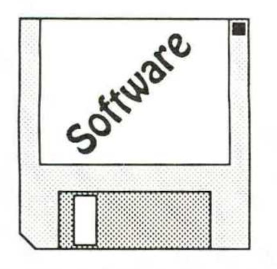
Bild 2: So ist der prinzipielle Aufbau von BS-HANDEL/3 zu verstehen: Funktionell eigenständige Module arbeiten "Hand in Hand".



Bild 3: Ein Beispiel für die Installationsprozedur: Für die verschiedenen Karteimasken sind die hier vorgestellten Feldnamen für branchenspezifische Anpassungen veränderbar.



Bild 4: Die Erfassung von Stammdaten. Zusätzlich zu den Adressangaben werden weitere Informationen in zusätzlichen Masken festgehalten.



Neuheiten

KatCe-ST DM 198,-

Bei KatCe-ST handelt es sich um ein integriertes Programmpaket, das ursprünglich für Einplatinencomputer mit der CPU 68000 entwickelt wurde. Bei der Programmierung wurde darauf geachtet, es später auch auf andere Computer übertragen zu können.
Ziel bei der Entwicklung war es, ein System zu schaffen, mit dem sowohl mittels einer Hochsprache als auch auf Maschinenebene schnell und effektiv programmiert werden kann.
Für eine schnelle Arbeil sollten alle Programmteile ständig im Speicher präsent sein, sie sollten gegenseitig schnell aufrufbar sein und es sollte genug Platz für Ouelltexte übrig bleiben. KalCe-ST enthält eine eigene „Shell" mit Maschinenmonitor, Disassembler, Editor, Assembler, Pascalcompiler und verschiedenen Bibliotheken (Real, VDI, AES, Grafik). Da es vollständig in Assembler (übrigens mit seinem eigenen) programmiert wurde, nimmt es nur einen Platz von ca. 84 KByte ein. Der Rest des Platzes bleibt dann für Quelltext und erzeugten Code.
Um diese Kompaktheit zu erreichen, wurde auf eine GEM-Umgebung verzichtet. Das heißt aber nicht, daß die erzeugten Programme auf die GEM-Annehmlichkeiten verzichten müssen.
Um möglichst viel Quelltext im Speicher unterbringen zu können, wird ein eigenes Textformat mit komprimierter Darstellung verwendet. KatCe-ST läuft auf allen ST-Modellen in allen Auflösungen.

Pascal

Der Compiler orientiert sich an vorhandenen Standards. So ist er weitgehend mit Turbo-Pascal 3.0 kompatibel. Darüberhinaus bietet er zahlreiche Erweiterungen. Während der Entwicklungsphase wird man den Code im Speicher erzeugen (und austesten), eine Speicherung auf Diskette ist dann erst zum Abschluß erforderlich.
Die Übersetzungszeit liegt bei etwa 400 Zeilen pro Sekunde.Weil diese Geschwindigkeit so hoch ist, kann man auch sehr lange Programme „im Stück" compilieren, deshalb kann man auf einen Linker verzichten. Übersetzte Assemblerteile werden während des Compilierens direkt eingebunden.

Merkmale

- Debuggen
- KatCe-St Pascal ermöglicht ein sehr einfaches Multitasking mit bis zu acht Prozessen
- GEMDOS, BIOS und XBIOS, AES und VDI
- Kat-Grafik wird unterstützt
- Editor (bildschirmorientierter Programmeditor ohne Mausunterstützung)
- Assembler
- AES-Funktionen
- Maschinenmonitor
- Bibliotheken

pro-CAM DM 189,-

Life-Simulation

Im Gegensatz zu herkömmlichen Simulationskonzepten gibt es eine neue Idee namens CAM. Cellular Automata Machines arbeiten auf rein graphischer Ebene. Sie sind heute so weit entwickelt, daß sie komplexe, naturwissenschaftliche Vorgänge simulieren konnen.
Erstmalig ist nun ein professioneller Cellularer Automat (CAM) für den ATARI erhältlich. Er bietet eine graphische Benutzerschnittstelle und ist daher gut geeignet, den Einstieg in diese neuartige Welt der Simulationssysteme zu vollziehen. Mit dem System können komplexe Probleme bearbeitet werden: Zu dem weitgefächerten Anwendungsspektrum von Cellulä ren Automaten gehören u.a. Simulationen von chemischen Reaktionen und physikalischen Prozessen. Ebenso können Fragestellungen der Ökonomie, der Soziologie und der Biologie behandelt werden.

Eigenschaffen von pro-CAM:

- benutzerfreundliche, mausgesteuerte Arbeitsoberfläche
- hohe Geschwindigkeit (bis zu 1 Generation pro sek bei einer Auflösung von 256 x 160 Pixel), da vollständig in Assembler programmiert
- Filmoption erlaubt bis zu 8 Generationen pro sek
- Farb- und S/W-Versionen sind eigenständige Programme mit jeweiligen Stärken
- Lieferumfang: 1 Buch, 1 Diskette mit Programmen, 4 Disketten mit Anwendungen aus Mathematik, Physik, Chemie, Soziologie und Geographie
- bereits bei mehreren europäischen Universitäten im Einsatz



ST Plotmal DM 298,-

Plotmal ist ein neues Programm für die grafische Präsentation und die Auswertung von Daten mit dem ATARI ST. Umfangreiche Optionen und eine speziell für dieses Programm entwickelte grafische Benutzeroberfläche ermöglichen die mathematische Bearbeitung und das Verändern und Einstellen aller vom Programm verwendeten Daten und Parameter sowie die flexible Gestaltung und Beschriftung der mit den Daten erstellten Diagrammen. Das WYSIWYG-Konzept gestattet dem Anwender, ein auf dem Bildschirm dargestelltes Diagramm voll auf seine Bedürfnisse anzupassen, um es dann auf Nadel- oder Laserdrucker bzw. Plotter auszugeben.

Merkmale

- Eigene grafische Benutzeroberfläche mit Dialogboxen
- Maussteuerung und umfangreicher Tastaturbefehlssatz
- Hilfe-Funktion
- Umfangreiche Pufferfunktionen
- Maximal 250 Datensätze und 9999 Wertepaare
- Gruppierung von Wertepaaren durch Marken
- Selektion von Datensätzen für die Anwendung verschiedener Optionen durch Ansprechen der gewünschten Marken
- Zwei völlig unabhängige Arbeitsebenen

Diagramme:

- X,Y-Diagramm- und Säulenchart-Option
- Lineare und logarithmische x- und y- Achsen
- Umrechnungsfunktion zur Manipulation der x- und y-Werte
- WYSIWYG-Darstellung der Daten vor der Peripherie-Ausgabe
- Ausgabe der Diagramme auf Nadel -, Laserdrucker oder Plotter
- Ausgabemodus DIN A3, DIN A4, Quer-, Hochformat

Grafische Symbole und Linien:

- 10 Linienarten, 30 Symbolarten
- Symbol- und Linienvorrat kann editiert und erweitert werden

Beschriftung:

- Freie Beschriftung im gesamten Bereich des Grafikausdruckes
- Verschiedene zuladbare Zeichensätze
- Beschriftung im proportionalen und nicht proportionalen Modus

Text- und Wertepaar- Editor

Auswerte-Funktionen

Export- und Import-Funktionen

- Importieren und Aufbereiten beliebiger ASCII-Dateien

BESTELL-COUPON

☐ Info-Broschüre kostenfrei

– Alle genannten Preise unverbindliche Preisempfehlungen –

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Bitte senden Sie mir:

		Best.-Nr.	Preis
_____	KatCe-ST	(S-9039)	à 198,- DM
_____	Pro-CAM	(S-9040)	à 189,- DM
_____	ST Plotmal	(S-9050)	à 298,- DM

zzgl. Versandkosten DM 6,- (Ausland DM 10,-) unabhängig von der bestellten Stückzahl

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Oder benutzen Sie die in ST-Computer eingeheftete Bestellkarte

In der Schweiz:
Data Trade AG
Landstr. 1
CH - 5415 Rieden-Baden

In Österreich:
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

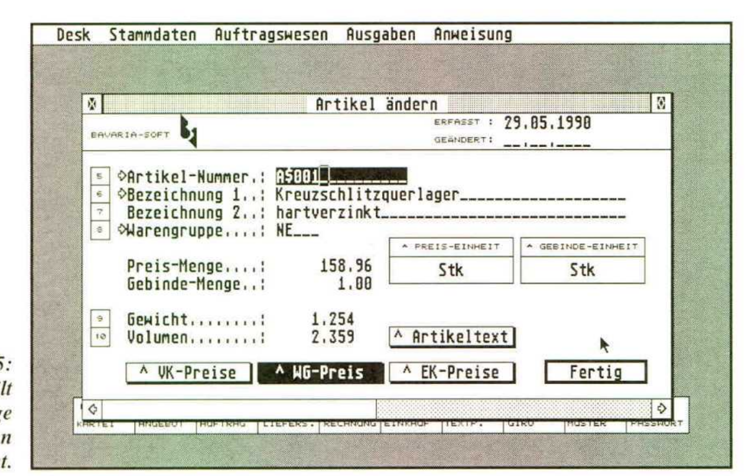


Bild 5: Für jeden Artikel hält man hier wichtige "technische" Daten fest.

Desk Stammdaten Auftragswesen Ausgaben Anweisung

Abruf: 90052920004 für Müller-Thurgau

A: 1 N: 1094042.07 R: 33836.35 S: 153165.89 B: 1247207.96

Pos.	Menge	Art.-Nr	Bezeichnung	Netto	Rabatt	Br
1	3.00	A$001	Kreuzschlitzqu	1094042.073	3383.63	1247207

Artikelübersicht

Nummer	Name	WG	Lagerort	Preiseh	Gebindeeh	Pre
A$001	Kreuzschlitzquerlage	NE	1000	Stk	Stk	

Bild 6: Hier wird ein Formular zusammegestellt (z.B. eine Rechnung). Die obere Liste zeigt die einzelnen Formularposten, die untere Liste eine Übersicht der bisher angesprochenen Artikel.

Desk Stammdaten Auftragswesen Ausgaben Anweisung

Abrufnummer für: 90052920004/M — Versandarten

^P MAHNSPERRE · ^ BARZAHLUNG · ^ UMSATZ BUCHEN · Spezial-Formulare · ^ Druckfreigabe

Abruf..........: 90052920004 erstellt am: 29.
◇Angebot ...: AN900529002 erstellt am: 29.
◇Rechnung an....: 10000 Name.......: Mül
Lieferung an...: 10000
Vertreter......:

^Warenwert......: 1094042.07
Rabatt.........: %
Versandkosten..:
Aufschlag......: %
Netto Summe....: 1094042.07
Steuer Summe...: 153165.89
Brutto Summe...: 1247207.96
Skontierfähig..: 1247207.96

Skon
Skonto - 2: %
rein netto:

Versandarten: A Post · B Nachnahme · C Express · D Paketdien. · E Frei Haus · F ab Werk · G Selbstabh. · H Bahn · I Bahn-Expr. · J Spedition

Zusatz-Text....:

Bearbeiten

Bild 7: Einzelne Faktoren für die Abwicklung eines Auftrags sind hier zusammengetragen.

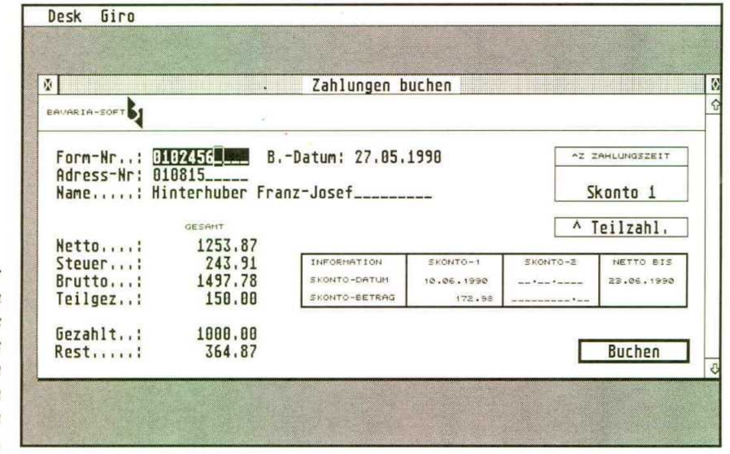


Bild 8: BSH/3-GIRO ist ein kleines Zusatzprogramm, das die Buchung von Zahlungsvorgängen und das Mahnwesen übernimmt.

Der Auftrag lacht

Wie wird denn nun ein Auftrag abgewickelt? Stammdaten zum Kunden liegen vor, Artikel sind erfaßt, Bestände festgestellt. Die Funktionstasten >F2< bis >F5< stellen fast alle Routinetätigkeiten in Beziehung auf den Kunden dar. >F6< (Einkauf) zeigt eine typische Handlung in Richtung eines Lieferanten. Die vollständige Auswahl an Tätigkeiten verbirgt sich im Menü "Auftragswesen". Zunächst wählt man die Adresse des Kunden. Dies geschieht in einer Suchmaske, die genauso aussieht wie jene bei der Kundenerfassung. Wichtig sind nun die Feldbezeichner mit einem kleinen Pfeil davor. Dies sind die Schlüsselfelder, und nur dort kann man ein Suchkriterium eingeben. Anschließend stellt sich die Frage, wie die Artikel der Kundenbestellung zusammengestellt werden sollen:

- per Anklicken in einem Übersichtsfenster (als Liste bestehender Einträge),
- aufgrund einer bereits bekannten Artikelnummer (weil sie der Kunde in seiner Bestellung schon aufgeführt hat) oder
- als nachträgliche manuelle Nacherfassung (wenn es diesen Artikel noch nicht in der Kartei gab).

Je nachdem, welche Verfahrensweise gewählt wird, öffnen sich verschiedene Fenster. Im 1. Fall öffnen sich nur zwei Fenster, in der oberen Hälfte das vorläufige 'Formular', in der unteren Hälfte die Artikelübersicht. Im 2. Fall kommt unten ein drittes Fenster hinzu, als Suchmaske, welches im 3. Fall als erweiterte Eingabemaske arbeitet. Steht nun der Artikel fest, wird er in das obere Fenster übertragen, gewissermaßen in einen Vorratsspeicher für den späteren Formularausdruck.

Wenn diese zwei Fenster geschlossen werden, erscheint eine Maske "Summen bearbeiten". Wenn der Artikel bisher nur 'in der Absicht stand', verkauft zu werden, bedeutet die Bearbeitung als Summe, daß die Einzelartikel zusammengerechnet und gleichzeitig als interner Umsatz von Lager (als Abgang) nach Versand gebucht werden. Dies löst gleichzeitig eine Sollstellung des Rechnungsbetrages in der Kundenstatistik aus. Gespeicherte Vorgänge werden alle in einen 'Sammelbehälter' bewegt, wo sie auf die wirkliche Verarbeitung (sprich Lieferschein- bzw. Rechnungsdruck) warten. Solange die Vorgänge in dieser Sammlung liegen, können sie insgesamt oder auch nur einzelne Positionen nachträglich verändert werden.

Auswertungen und Listen

In einem separaten Menü sind verschiedene betriebswirtschaftliche Informationen abrufbar. So zeigt eine Kundenstatistik die monatlichen und jährlichen Umsatzsummen für einen oder mehrere Kunden. Ähnlich ist die Aufstellung für Lieferanten zu verstehen. Eine Artikelstatistik läßt Rückschlüsse auf Verkaufserfolge und Rohgewinne zu. Eine Fakturstatistik vermerkt alle bisherigen Vorgänge (Aufträge, Lieferungen, Rechnungen, Einkäufe usw.) für fest vorzubestimmende Zeiträume. Auch eine ständige Inventur und Lieferrückstandsmeldungen könnten aus diesem Programmteil kommen.

Irgendwann im Verlaufe der Arbeiten wird es nötig sein, die Ergebnisse zu Papier zu bringen. Dabei stellt BS-Handel/3 einen sogenannten Listengenerator zur Verfügung. Er bereitet die Daten in Verbindung mit Musterformularen so auf, daß diese komplett an die Textverarbeitung übergeben werden können.

Das Programm unterscheidet zwischen "Briefen" und "Listen". Briefe sind im Grunde jene 'Textaufbauten', deren Aussehen schon durch den Globaltext gestaltet wurde. An bestimmten Stellen aber kann noch nicht feststehen, welche individuellen Textteile hinzukommen, dann sind dort sogenannte Platzhalter ersatzweise eingefügt. Deutlich wird dies am Beispiel eines Serienbriefes: Der Globaltext sind all jene Stellen, die in jedem Brief gleichbleiben. Individualtext sind jene Passagen, die sich von Adressat zu Adressat ändern. Die Platzhalter, in aller Regel die Feldnamen, kennzeichnen all jene Positionen, die durch Feldinhalte später ersetzt werden (z.B. Name, Vorname, Straße usw.). Die Abarbeitung eines Briefes wiederholt sich solange, und zwar vollständig, wie noch (meist vorher selektierte) Datensätze zur Verfügung stehen.

Unter Listen versteht BS-Handel/3 Textanordnungen, die in ihrer Anzahl stark variieren können. Wenn ein Kunde 5 verschiedene Artikel bestellt hat, stehen eben die 5 untereinander. Bei anderen Kunden entsprechend mehr oder weniger. Hierbei wiederholt sich die Abarbeitung innerhalb eines Formulars so lange, wie gleiche Komponenten (hier: Artikelangaben) zur Verfügung stehen. Auch hier gibt es Platzhalter, die die einzelnen Positionen bezeichnen. Platzhalter müssen aber nicht nur Feldnamen sein, die stellvertretend diese Druckposition einnehmen, es sind auch ausführliche Formeln oder Entscheidungsanweisungen möglich.

Die Bereiche Mahnwesen und Zahlungen wurden mit Absicht in ein kleines Separatprogramm gepackt. Der Grund liegt darin, daß beim Bearbeiten der Buchungen mit BS-FiBu/3 andere Vorgänge ablaufen, als wenn nur das Programm BS-Handel/3 zur Verfügung steht.

Im Ganzen betrachtet

Ich will ganz ehrlich sein: BS-Handel/3 stellt eine Vielzahl an Funktionen zur Verfügung, die ich unmöglich in der (verhältnismäßig) kurzen Zeit abschätzen konnte. Auch wäre es unmöglich, alle Vorgänge hier auch nur aufzuzählen. Wichtig ist allein die Tatsache (und das kann man mit ein paar wenigen Artikeln und Adressen auch schon sehen), daß das Konzept logisch aufgebaut, durchgängig und plausibel ist. Nach kurzer Einlernphase wird es dem ersnthaften Benutzer möglich sein, seinen kompletten Kundenstamm und das Lager zu verwalten. Es gibt kaum einen Punkt zur Beanstandung (jedenfalls habe ich keinen dramatisch schlimmen gefunden). An ein paar Kleinigkeiten könnte allerdings noch gefeilt werden.

Beispiele: der Wechsel zwischen Funktionstasten und Menüs bei "Kartei"; der Wechsel in ein externes Textprogramm beim Ausdruck.
Schön ist zum Beispiel, daß alle Bedienungspunkte nicht nur per Menü, sondern immer auch mit Tastenkombinationen erreichbar sind. Die Verarbeitungsgeschwindigkeit soll gegenüber der Version 2 um mindestens 400% beschleunigt worden sein (was ich mangels Vergleichsobjekt leider nicht überprüfen konnte). BS-Handel/3 läuft auch auf dem ATARI-TT und soll dort noch einmal um 500% schneller sein - quod erat demonstrandum.

Das Handbuch hat mir ebenfalls gut gefallen. Die 235 Seiten beinhalten viel Information und sind ausreichend bebildert. Die Erklärungen sind ausführlich, und in dem gestalterischen Aufbau findet man sich leicht zurecht. Es ist flüssig und stellenweise etwas fröhlich geschrieben. Besonders die Kapitel zum Listengenerator und einigen Anwendungsbeispielen sind sehr ausführlich und leicht nachvollziehbar.

Ein getrenntes Referenzhandbuch zeigt noch einmal in knapper Form alle Funktionselemente.

BS-Handel/3 kostet 698,- DM. Der Preis bewegt sich durchaus im vertretbaren Rahmen. Es existiert ein Update-Service für die Vorgängerversionen.

DK

Bezugsquelle:

Bavaria-Soft Datentechnik GmbH
Otoo-Hahn-Straße 25
8012 Ottobrunn bei München
Tel.: 089/6097838











It is MEGA!

Das Vektor- und Pixel-Grafikprogramm für den gehobenen Anspruch. Hier wird exaktes Konstruieren und Zeichnen mit dem ATARI ST zum reinen Vergnügen. Wahre Freude bereitet dem Anwender die Möglichkeit, z.B. seine mit MegaPaint Professional erzeugten Grafiken nicht nur einfach auszudrucken, sondern auch in Calamus zu plazieren und anschließend mit der Präzision eines PostScript-Belichters auszugeben. Umfangreiche Features und Einstellmöglichkeiten überlassen nichts dem Zufall:

- Arbeiten und automatisches Bemaßen nach DIN-Norm
- Vektorisieren von gescannten Vorlagen
- direkte Ansteuerung diverser Scanner
- 7680 x 7680 Pixel interne Auflösung
- stufenloses Zoomen von 1-900%
- umfangreiche Blockfunktionen
- Signum!-Zeichensatzkonverter
- abschaltbare Snap-Funktion
- mehrere Bildebenen
- Symbolbibliotheken

u.v.m.

MegaPaint Professional

DM 799,-

MegaPaint

(ohne Vektormodul und Signum!-Zeichensatzkonverter)

DM 499,-

MAXON Computer GmbH
Schwalbacherstr. 52
6236 Eschborn
Tel.: 06196/481811

MAXON computer gmbh

M.V. ZIMMERMANN & M.F. LOWACK

PC - Computer und mehr...

Ein leistungsfähiger Partner stellt sich vor

Wer ist PC?

PC wurde 1989 in Regensburg gegründet. Heute gibt es im gesamten Bundesgebiet bereits 30 Niederlassungen mit Flächen von 200 bis 1 500 m². In den nächsten Jahren wird **PC** sein Filialnetz weiter ausbauen. Der Vorteil für Sie: Immer kürzere Wege zu Ihrem Computer-Fachmann.

PC wendet sich gleichermaßen an den professionellen wie an den privaten Interessenten. Das Programm reicht vom klassischen Computer, seiner Peripherie und Software bis hin zu Produkten der Telekommunikation und Bürotechnik. Darüber hinaus unterhält **PC** eine Reihe eigener Schulungszentren, in denen Sie Kurse und Seminare zu vielen EDV-Themen besuchen können.

Die Entstehung von PC

Die Entstehungsgeschichte von **PC** ist eng verbunden mit den Elektro-Fachmärkten **ProMarkt, UniMarkt und Elektroland**.

1976 eröffnet der erste **ProMarkt** in München-Gräfelfing. Seit 1983 gibt es im ProMarkt eine Computer-Abteilung. 1985 hat sie einen Umfang angenommen, der die Gründung eines separaten Computer-Hauses notwendig macht. Dieser Stand-Alone heißt **ProCE** und bietet höherwertige Computertechnologie ebenso wie Vortragsreihen und Computerkurse.

Bereits zu dieser Zeit werden durch die heutige **PC**-Geschäftsleitung Beratung und Schulung über den Kauf hinaus in die Tat umgesetzt. Die ersten User-Clubs und Arbeitsgruppen entstehen. Betriebliche Umstrukturierungen bei ProMarkt führen zur Auflösung von ProCE. Im **UniMarkt** Augsburg werden die Schulungsprogramme im Computerbereich weiterentwickelt. Heute finden sie vielfach erprobt in den **PC**-Schulungszentren ihre Anwendung.

Einen weiteren bedeutenden Einfluß auf die Entwicklung von **PC** nahm **Zimmermann Elektroland** Regensburg. 1973 gegründet und 1979 zum Fachmarkt für Unterhaltungselektronik ausgebaut, wird 1984 das Sortiment um den Bereich Home-Computer erweitert.

Nach der Übernahme von Zimmermann Elektroland durch einen großen deutschen Handelskonzern sind die finanziellen Voraussetzungen gegeben, die ProCE-Idee deutschlandweit zu realisieren: ein kundenorientiertes, ganzheitliches Vertriebssystem der Informationselektronik.



PC entsteht.

Die einstige Home-Computerabteilung im Elektroland ist längst den Kinderschuhen entwachsen und hat bereits einen ernstzunehmenden professionellen Standard erreicht. Der wesentliche Teil der **PC**-Führungsmannschaft findet sich hier zusammen. Externe Kräfte aus Industrie und Handel ergänzen dieses durch jahrelange Zusammenarbeit gut eingespielte Team in optimaler Weise.

Ein Unternehmen entwickelt sich.

In einem ersten Schritt wird das Elektroland in Nürnberg zum Testmarkt für die **PC**-Idee ausgewählt. Der Bereich Unterhaltungselektronik wird zugunsten des erweiterten Computerangebotes aufgegeben. Das Nürnberger Experiment verläuft äußerst erfolgreich und ebenso der bei Evertz Düsseldorf eingerichtete Computer-Shop. Überzeugende Argumente, **PC** als selbständigen Vertriebsbereich großflächig aufzubauen. Und zwar genau nach diesen beiden Strukturen: einmal als reine Computer-Häuser und einmal nach dem Shop-in-Shop-Prinzip.

Warum PC?

Bei der Namenssuche wird von Anfang an **»PC«** favorisiert. Kurz und prägnant weist es auf das Kernprodukt der gesamten EDV hin, den Personal Computer. Andererseits steht es für die Leistungsfähigkeit von **PC, »Professional Competence«**.

Was bietet Ihnen PC?

Der **PC**-Gedanke ist logisch und einfach zugleich: die Zusammenfassung aller vorhandenen Vertriebssysteme unter Ausklammerung der jeweiligen Nachteile. So finden Sie bei **PC** ein umfassendes Sortiment, Markenware, eine kompetente Beratung und günstige Preise. Zudem besitzt jedes größere Computer-Center eigene Schulungsräume. Aus einer Vielzahl von Computer-Kursen wählen Sie hier den für Sie passenden aus. Alle für den erfolgreichen Einstieg in die EDV nötigen Voraussetzungen können Sie so mit einem einzigen Partner schaffen.

Schulungen

Auf sein Schulungsprogramm ist **PC** besonders stolz. Das Hauptziel ist es, die eigenen Mitarbeiter optimal aus- und weiterzubilden. Problem-Analyse, Lösungsfindung und die Betreuung nach dem Kauf sind dadurch für Sie optimal gewährleistet. **PC** wendet für die Mitarbeiter-Schulung jährlich eine sechsstellige Summe auf. Eine enorm hohe, aber lohnende Investition, denn die hochgesteckten Ziele von **PC** lassen sich nur mit motivierten und qualifizierten Mitarbeitern erreichen.

Aber auch für Sie als Anwender hält **PC** ein umfangreiches Kursprogramm bereit. Wenn Sie alles über Ihren Computer oder Ihr Programm wissen wollen (oder müssen), können Sie aus den einzelnen Modulen beliebige Kombinationen zusammenstellen und Ihre Interessen gezielt abdecken. In kostenlosen Vorträgen zum Thema Computer sowie in Computer-Grundlagen- und Schnupperkursen werden Sie ohne Kaufzwang an den Umgang mit dem Computer herangeführt. Einsteigerkurse für sinnvolle EDV-Anwendungen sowie weiterführende Seminare zu vielen Geräten und Programmen schließen sich an.

Einige **PC - Computer-Center** unterhalten eigene User-Clubs, die den Ideen- und Erfahrungsaustausch im Kreise Gleichgesinnter fördern.

Spezialisierung

Ein wohldurchdachtes Sortiment von Hardware, Programmen und EDV-Zubehör bildet die Basis für umfassende Computerlösungen. Der Einsteiger-Computer für den Privathaushalt ist bei **PC** ebenso erhältlich wie die professionelle Ausstattung großer Firmen und Behörden mit Mehrplatzsystemen und Produkten der Bürokommunikation. Das wirkliche Erfolgsgeheimnis aber liegt im mehrschichtigen Vertriebssystem. Mitnahmeartikel bei einfacher Ware, kompetente Beratung bei höherwertigen Produkten und Betreuung durch den Außendienst bei komplexen Systemen.

Preisvorteile durch zentralen Einkauf

PC kauft Standard-Produkte direkt beim Hersteller. Die Eigenmarken **PROFI DATA** und **IQ** werden nach genauen Spezifikationen eines **PC** - Experten-Teams gefertigt. Die durch die Umgehung von Zwischen- und Großhändlern erzielten Preisvorteile gibt **PC** in vollem Umfang an Sie weiter. Eine umfassende Betreuung ohne überzogene Preisforderung wird dadurch erst möglich.

Kundendienst und Support

Erfahrene Spezialisten bürgen für einen reibungslosen und schnellen Reparatur-Ablauf. Ein großzügig geplanter Gebäudekomplex für diese wichtige Aufgabe ist momentan im Bau. Der Umzug in die neuen Räume ist für September '90 geplant. Damit können auf Jahre hinaus kürzeste Durchlaufzeiten mit hoher Arbeitsqualität erreicht werden. Zur Behebung von kleinen Service-Aufgaben direkt vor Ort sind die größeren **PC** - Computer-Center mit geeignetem Personal und Service-Räumen ausgestattet. Die Service-Leistungen werden durch die regional im Aufbau befindlichen Außendienste ergänzt.

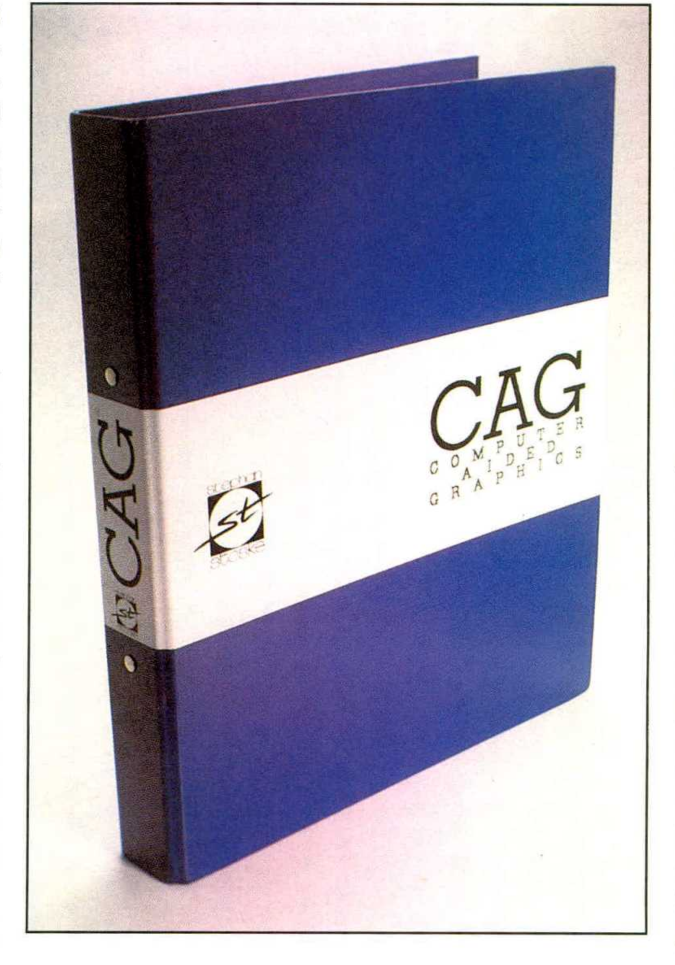
Geprüfte Technik bei PC

Alle neuen Produkte werden vor der Aufnahme ins Sortiment ausführlich getestet. Als **PC**-Kunde erhalten Sie Geräte, die erprobt sind und auch nach Jahren noch zuverlässig ihren Dienst tun.

Viele Extras und Bedienungskomfort

Computer Aided Graphics - CAG

Sicher sagen Sie jetzt: "Was soll ich denn schon wieder mit einem neuen Grafikprogramm - ich hab' doch schon eins..." Und schon liegen Sie völlig falsch. Nicht nur Funktionsvielfalt und reichlich Extras stehen bei CAG im Vordergrund, sondern auch Benutzerfreundlichkeit, Komfort und Übersichtlichkeit.



Mit CAG lassen sich Pixelgrafiken erstellen oder bearbeiten. Aber für alle Fälle ist auch ein kleiner Vektorteil enthalten, der zwei- und dreidimensionale Grundfunktionen erledigt. Angesichts der Länge des Programmes mußte nach Angaben des Autors auf ausgiebige Vektorbearbeitung verzichtet werden. Dafür wird der Käufer mit zahlreichen Grafikfunktionen belohnt, und auch das Handbuch ist vorbildlich in puncto Umfang, Verständlichkeit und Zusatzinformation. Doch was reden wir lange um den heißen Brei herum, kommen wir zur Sache.
Das rund 120 Seiten umfassende Handbuch wird im festen DIN A4-Ordner geliefert, der auch einen Disketteneinschub besitzt. Es ist professionell gedruckt und in sieben Kapitel unterteilt.

ATARI-Menü

Reich gespickt ist unsere Grafik-Software mit Menüs und Untermenüs, die sich zum größten Teil auch durch Tastenkommandos aufrufen lassen (siehe Bild 2). Bereits im ATARI-Info-Menü hebt sich CAG von anderer Software ab. Hier wird nicht nur die Arbeitszeit angezeigt, sondern der User hat die Möglichkeit, das Programm über Eingabe eines Paßwortes zu sperren. Danach erscheint solange ein schwarzer Bildschirm, bis das Paßwort erneut eingegeben wird. Diese Funktion erlaubt es, das Programm samt aller Daten vor dem Zugriff durch unberechtigte Personen zu schützen.

In diesem Pulldown-Menü kann der Benutzer sämtliche Peripherie-Ein- und Ausgaben durchführen, wie zum Beispiel Laden und Speichern von Blättern, Grafiken und Objekten. Erwähnenswert ist die Palette von Dateiformaten, die CAG einladen kann. Was aber auf jeden Fall noch fehlt, ist das *.PAC-Format aus STAD, das mittlerweile auch zum Standard gehört. Lädt man nur einen Bildschirm ein, wird ein verschiebbares Rechteck angezeigt, dessen vier Ecken am Rand dargestellt werden. Somit ist ein punktgenaues Einpassen der zu ladenden Grafik möglich (siehe Bild 3). Beim Drucken besteht die Auswahl zwischen Einzel- und Massendruck. Standarddruckertreiber sind vorhanden für HP-Laserjet, ATARI-Laser SLM 804 und NEC P6 (180x180 dpi). 360x360er- und 9-Nadleranpassungen sind laut Angaben des Herstellers bis zum Erscheinen dieses Testberichtes kostenlos erhältlich. Auf Wunsch würde aber für registrierte Kunden auch jeder andere Drukker innerhalb von 14 Tagen angepaßt.

Eine interessante Option ist der Druck mit Rastern. Hierbei wird - im Gegensatz zu Calamus - kein Raster über den Druck gelegt, um den Ausdruck grober zu ma-



Bild 2: Das Hauptmenü CAG ist reich gespickt mit Untermenüs.

chen, sondern das angeklickte Raster wird vom Ausdruck "abgezogen". So lassen sich Ausdrucke aufhellen, was etwa beim Siebdruck erforderlich sein kann. Beim Menüpunkt *Massendruck* lassen sich bis zu 46 Dokumente ("Blätter") mit maximal 99 Kopien pro Blatt ausdrucken. Dies ist mit einer Drucker-Warteschlange zu vergleichen. Auf Knopfdruck kann CAG sogar sämtliche Blätter suchen, die auf dem aktuellen Laufwerk vorhanden sind (egal, ob inner- oder außerhalb jeglicher Ordner), und fügt diese automatisch in die Druckliste ein.

Der Menüeintrag *Diskette* besitzt sechs Untermenüs. Hier kann sich der Benutzer den aktuellen Speicher- und Diskettenplatz anzeigen lassen, Dateien umbenennen, löschen, Ordner erzeugen oder löschen und neue Disketten formatieren. Was jedoch fehlt, ist eine Einstellmöglichkeit des aktuellen Laufwerkes. So läßt sich etwa der Menüpunkt *Disketteninfo*, der belegte und freie Kapazität auf dem Datenträger und im Speicher anzeigt, nur auf das Startlaufwerk anwenden.



Bild 1: Durchnumerierte Alert-Boxen erleichtern die Suche im Handbuch.

Werkzeug

Sämtliche Zeichengrundfunktionen finden sich im Menü *Werkzeug* wieder. Mit Stift läßt sich freihandzeichen - entweder mit einem punktförmigen oder durch ein einstellbares Muster vorgegebenen Stift. Beim Pinsel kann der Zeichner zwischen den vier Größen sehr klein/klein/mittel und groß wählen. Mit dem Pinsel wird ebenfalls freihandgezeichnet, jedoch mit dem eingestellten Füllmuster. Die Funktion *Strich* zeichnet - wie bei anderen Grafikprogrammen - "ganz normale" Linien zwischen zwei Punkten. Auch der Menüpunkt *Strahlen* müßte bereits von anderer Software her bekannt sein.

Interessanter dagegen ist das Menü *Vieleck*, in dem der Grafiker die Unterauswahl zwischen Vieleck und Vektor hat. Mit Vieleck läßt sich ein ganz normales Polygon zeichnen, das danach am Bildschirm als Pixelgrafik zu sehen ist. Beim Vektor dagegen wird zunächst ebenfalls ein Vieleck gezeichnet, dessen Punkte einen zweidimensionalen Vektor ergeben. Dieser läßt sich in einem Vektor-Editor Punkt für Punkt edieren, vergrößern und verkleinern und um die drei Raumachsen X, Y und Z drehen. Auf die Funktionen Füllen, Kreis, Ellipse, Rechteck und Quadrat wollen wir aus Platzgründen nicht näher eingehen, da ihre Arbeitsweise von anderen Programmen her hinreichend bekannt sein müßte. Beim Sprühen stehen dem Anweder dreierlei rechteckige und drei runde "Sprühdosengrößen" zur Verfügung. Es kann sowohl mit Schwarz und Weiß, aber auch mit einem Muster gesprüht werden, die Sprühdichte ist veränderbar. Gut durchdacht ist der Schnell-Anwahl-Knopf "??", mit dem man ohne zusätzliches Anfahren des Pulldown-Menüs direkt zur Füllmustereinstellung umschalten kann.

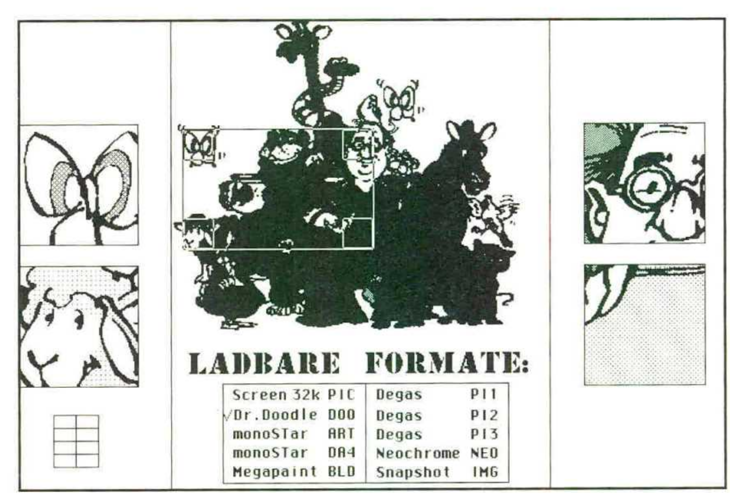


Bild 3: Geladene Bilder werden punktgenau eingesetzt.

Beim Bewegen kann aus viererlei verschiedenen Modi gewählt werden: rechteckig, elliptisch, Schere und Auto. Die ersten beiden erklären sich von selbst, unter Schere kann der Ausschnitt durch einen Polygon-Linienzug bestimmt werden. Interessant ist vor allem die Auto-Option. Hiermit lassen sich alle zusammenhängenden Punkte ausschneiden, die sich mit einer Füllfunktion in einem Schritt erreichen lassen würden. Der Radiergummi läßt sich auf eine bestimmte rechteckige Größe aufziehen, mit der dann radiert wird.

Die Lupe

Besonderer Hervorhebung bedarf die Lupe (siehe Bild 4). Hier lassen sich sechs verschiedene Auflösungen des Ausschnittes einstellen, von vierfacher bis hin zu

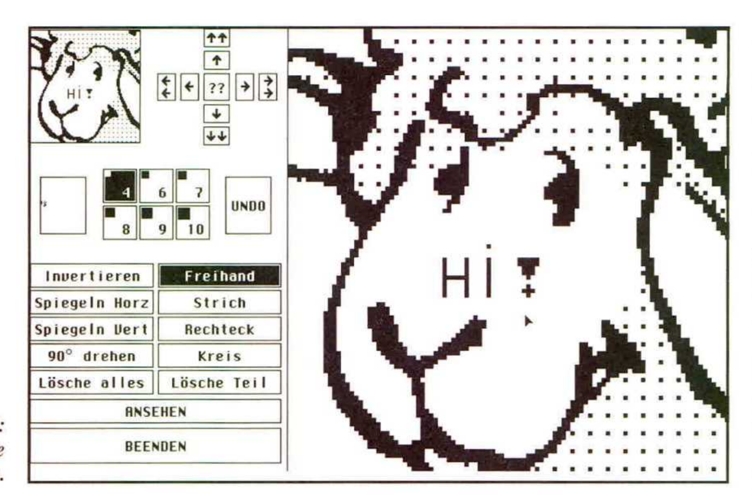


Bild 4:
Die Lupe bietet viele nützliche Funktionen.

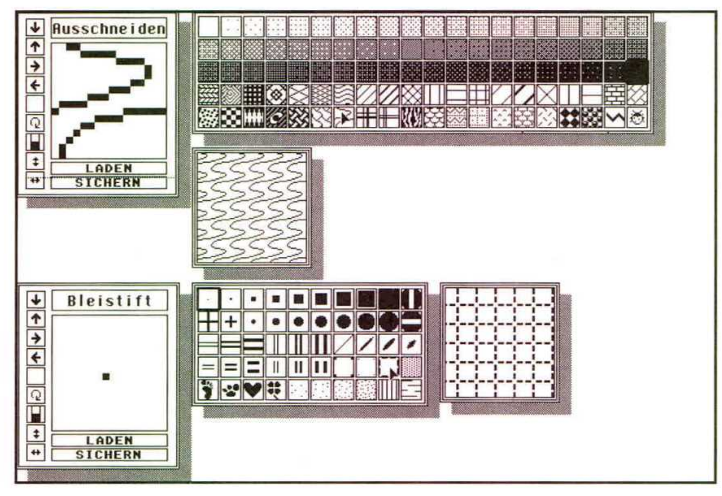


Bild 5:
Mit 50 Stift- und 100 Füllmustern hat der Zeichner genau die richtige Auswahl...

zehnfacher Vergrößerung gegenüber dem Original. Mit den Pfeil-Knöpfen oben links läßt sich der sichtbare Bereich der Lupe verschieben. Links neben der Vergrößerungseinstellung zeigt ein sogenannter "DIN A4-Locator" an, wo sich die Lupe auf dem Arbeitsblatt befindet. Durch Anwahl des "??"-Knopfes läßt sich der Ausschnitt proportional zur Mausbewegung auswählen. In der Lupe stehen verschiedene Funktionen zur Verfügung: Der Bereich läßt sich invertieren, spiegeln, um 90° drehen oder auch löschen. Ferner kann der Benutzer wählen zwischen Freihand, Strich, Rechteck und Kreis. Praktisch ist es, daß bei den Zeichenfunktionen in der Lupe nicht immer zwischen den Arbeitsfarben Schwarz und Weiß umgeschaltet werden muß (siehe Degas), sondern mittels linker Maustaste wird in schwarzen, mittels rechter Taste in weißen Punkten gezeichnet.

Einstellmöglichkeiten

Bei CAG stehen dem Anwender beinahe alle möglichen Einstellungen zur Verfügung. Neben "normal üblichen" Parametern für Liniendicke, -anfang, -ende usw. lassen sich außer Stift- und Füllmuster auch der Kreisausschnitt und die Flächenart variieren. Erwähnenswert sind die 100 verschiedenen Füllmuster, die zur Verfügung stehen. Davon sind im Standardzustand bereits drei Fünftel mit Graustufen-Mustern belegt. Dem Zeichner stehen also 60 (sechzig!) verschiedene Graumuster zur Verfügung, die fein abgestuft sind (siehe Bild 5). Vorbildlich!

Über eine abspeicherbare Datei lassen sich beliebig viele weitere Tabellen mit jeweils 100 Füllmustern erstellen. Mittels Laden und Sichern kann der Zeichner im Programm zwischen beliebig vielen Füllmustertabellen umschalten. An Stiftmustern stehen 50 verschiedene zur Auswahl, die sich ebenfalls verändern und abspeichern lassen. Problemlos kann eine Musterbibliothek angelegt werden. Bei Flächenarten besteht durch eine übersichtliche Box die Wahl zwischen Leer, Weiß, Schwarz und Muster und die Umrahmung ein- oder auszuschalten. Ferner kann man eckige und abgerundete Ecken einstellen. Wem das Löschen einzelner oder mehrerer der acht Bildschirme mittels Radiergummi zu umständlich ist, der kann durch die Funktion *Bereich löschen* einzelne Bildschirme oder das gesamte Arbeitsblatt per Knopfdruck löschen. Bei der Cursor-Form stehen mehrere Auswahlmöglichkeiten zur Verfügung: Passer, Kreuz, Kreuz mit Punkt und Punkt bieten für jede Gelegenheit den richtigen Cursor.

Zusätzlich läßt sich für Rechtecke etc. ein Fadenkreuz ein- oder ausschalten. Für exaktes Arbeiten kann auf die Mausbewegung ein Raster aufgelegt werden. Es existiert eine Rasterung für Zentimeter und Millimeter.

Wer sich durch die Töne gestört fühlt, die das Programm als Signalfunktion benutzt, kann die drei verschiedenen Akustikarten einzeln ausschalten. Schön gelöst ist der sogenannte Sicherheitsfaktor des Programmes. Nach einer einstellbaren Zeit zwischen fünf Minuten und zwei Stunden erinnert das Programm den Benutzer daran, seine Arbeit abzuspeichern. Im Gegensatz zu manchen anderen Programmen, die automatisch nach einer gewissen Zeit abspeichern, kann der Grafiker selbst entscheiden, ob er abspeichern möchte oder nicht. Sollte das zuletzt gespeicherte Arbeitsblatt doch noch gebraucht werden, wurde es nicht gerade unerwünschterweise automatisch überschrieben... Durch den SCRAP-Speicher kann beim nächsten Programmstart optional das zuletzt bearbeitete Arbeitsblatt nachgeladen werden. Der Menüeintrag *sichern...* dient dazu, die Programmparameter abzuspeichern. Es werden über siebzig Parameter abgespeichert. Diese erstrekken sich von den elf Dateipfaden bis hin zur Nachkommastellenzahl beim Bemaßen und werden im Handbuch ganz genau beschrieben.

Extra

Beinahe zur Standardfunktion in jedem Grafikprogramm ist der Bogen geworden, der zwischen zwei Punkten gezogen wird. Logischerweise findet er sich in unserem CAG wieder. Zusätzlich kann ein Bezier-Vieleck gezeichnet werden. Hierzu legt der Zeichner mittels Mausklicks mehrere Punkte fest, die von einer Bezier-Kurve durchzogen werden. Vor der Übernahme der Linie bietet CAG an jeder Ecke einen sogenannten "Magnetpunkt" an, mit dem die Kurve in ihrer Gestalt verändert werden kann. Für die Bemaßung von Strecken steht ebenfalls ein Menüeintrag zur Verfügung. Damit können Schriftrichtung, Zeichensatzgröße und Nachkommastellenzahl des Bemaßungstextes eingestellt werden (siehe Bild 6). Ferner hat der Grafiker die Wahl zwischen einer Angabe in Zentimetern, Millimetern oder keiner. Daß die Strichart (Muster, Linienanfang und -ende) einstellbar ist, ist selbstverständlich.

Das Programm besitzt eine Schnittstelle zum Programm *Graphbase* mit dem sich Grafiken namentlich besser verwalten lassen. Hierauf möchten wir aus Platzgründen nicht näher eingehen, auch des-

halb, weil es sich um ein eigenständiges Zusatzprodukt handelt. Das Programm CAG bietet dem Benutzer eine kleine Objektverwaltung (siehe Bild 7), mit der sich bis zu 999 Objekte per Nummer verwalten lassen. Es ist möglich, sich eine Objekt-Bibliothek anzulegen, die gepackt auf Datenträger abgespeichert wird. Natürlich lassen sich Objekte genauso einzeln abspeichern oder hinzuladen.

CAG und Vektoren

Das Programm besitzt zweierlei Menüeinträge für das Arbeiten mit Vektoren. Der Anwender kann zwischen zwei- und dreidimensionalen Vektoren wählen und diese nach Wunsch verändern. Im zweidimensionalen Teil stehen gleichzeitig maximal 20 Vektoren mit jeweils höchstens 128 Punkten zur Verfügung. Der Benutzer hat die Möglichkeit, seine Vektoren Punkt für Punkt zu verändern und kann das Objekt um die drei Achsen X, Y und Z drehen, das Objekt zentrieren lassen oder die Größe verändern (siehe Bild 8). Ist der gewünschte Zustand hergestellt, kann der Vektor als Pixelgrafik in das Arbeitsblatt übernommen werden, wahlweise schwarz, weiß, leer oder mit einem Muster gefüllt.

Bei den dreidimensionalen Vektoren wurden die Möglichkeiten aus Speicherplatzgründen ein wenig eingeschränkt. Zwar läßt sich das Objekt ebenfalls um die drei Achsen beliebig drehen, und die Größe ist variierbar, aber es stehen dem Benutzer nur zehn verschiedene Vektoren zur Verfügung. Diese sind vom Programm fest vorgegeben und lassen sich nicht verändern.

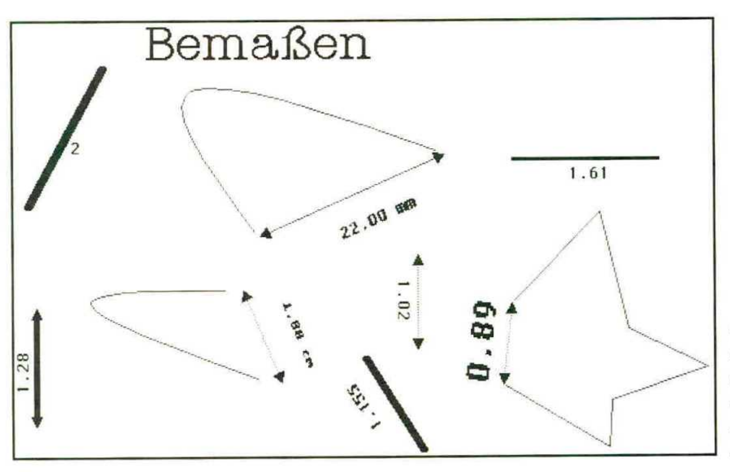


Bild 6: Halbautomatisches Bemaßen ist kein Problem, nur die Drehroutine ist noch nicht perfekt.

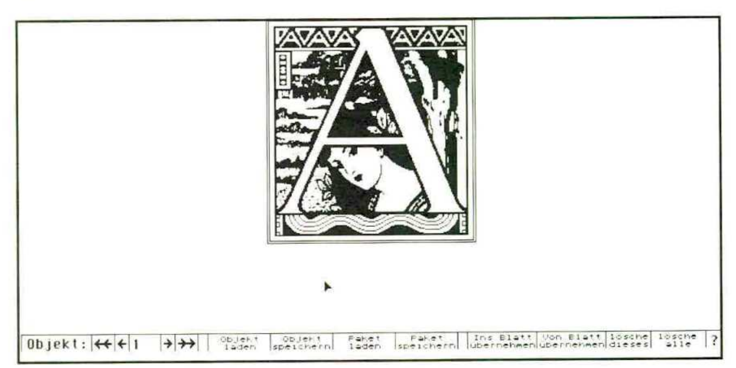


Bild 7: Es lassen sich bis zu 999 Objekte verwalten.

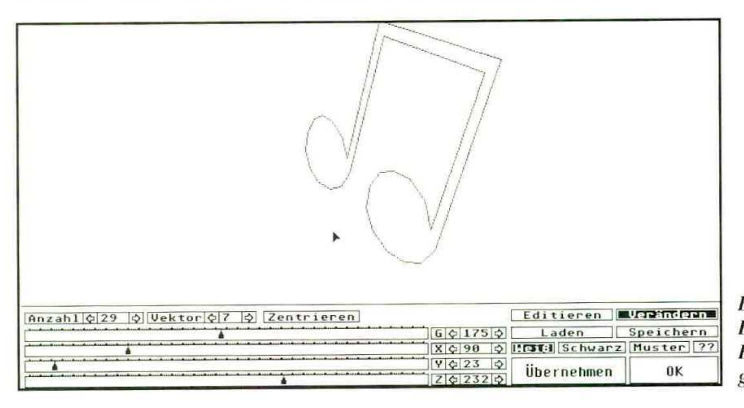


Bild 8: Ein Mini-Vektor-Editor ist mitunter ganz nützlich...

Spezial

In diesem Menü tauchen die Sonderfunktionen auf. Damit kann zum Beispiel ein n-Eck mit drei bis zwölf Ecken problemlos gezeichnet werden. Interessant ist vor allem die Möglichkeit, Grauflächen zu erzeugen (siehe Bild 9). Es gibt dreierlei Arten: zufällig, abgestuft und mit Muster. Bei der Einstellung "zufällig" können Sie ein Polygon (Vieleck) aufziehen, das danach zu 50 Prozent auf der gesamten Fläche gleichmäßig mit Punkten gefüllt wird. Am interessantesten ist die Funktion "abgestuft". Es können Verlaufswerte zwischen minus 100 und plus 200 Prozent angegeben werden, womit sich innerhalb eines Vierecks Grauverläufe erzeugen lassen. Durch Angabe von Werten, die kleiner als 0 und größer als 100 Prozent sind, erreicht man eine breitere weiße beziehungsweise schwarze Fläche.

Ein äußerst nützliches Extra ist das Transponieren. Hiermit lassen sich beliebige Objekte mit einem Hintergrund "füllen". Dazu schneidet man das Objekt aus, füllt die Teile, die mit einem Hintergrund unterlegt werden sollen, und sucht dann den Hintergrund aus. Gerade durch die Funktion *Transponieren* lassen sich sehr hübsche Effekte erreichen (siehe Bild 9). Interessant sind auch die beiden Pinselarten *Japan* und *Bürste*. Beim japanischen Pinsel zieht der Cursor in einem einstellbaren Abstand einen zweiten Punkt hinterher, der genau den Bewegungen des Cursors folgt, nur ein wenig zeitversetzt. Dadurch entsteht eine Art Pendel. Zwischen den beiden Cursor-Punkten zieht CAG jeweils eine Linie, wodurch recht schöne "Malereien" hervorgezaubert werden können. Die Bürste dagegen, die sich in den drei Arten senkrecht, waagrecht und diagonal einstellen läßt, dient in erster Linie zum Zeichnen von Landschaften - insbesondere von Rasenflächen oder Sträuchern. Die Zeichenfunktion Fluchtpunkt zeichnet ein Schachbrett, das auf einen bestimmten Punkt im Raum zuläuft. Diese Programmoption werden Sie sicher nur äußerst selten benutzen, zumal sie in unseren Augen eher eine kleine Spielerei darstellt.

Mit den beiden Funktionen *Verformen* und *Kanten* lassen sich Grafiken in ihrer Gestalt ändern. Beim Verformen hat man die Wahl zwischen proportional, unproportional und unförmig. Hinter der Option proportional verbirgt sich eine Vergrößerungs- beziehungsweise eine Verkleinerungsroutine. Hier wird die Veränderung bei beiden Achsen (X und Y) um den gleichen Faktor durchgeführt. Bei unproportionaler Verformung läßt sich das Objekt dagegen auch in die Höhe oder Breite ziehen oder stauchen. Die Wahl von unförmig steckt das Objekt in ein Viereck, dessen vier Ecken beliebig verzogen wer-



Bild 9: Graufläche und Transponieren

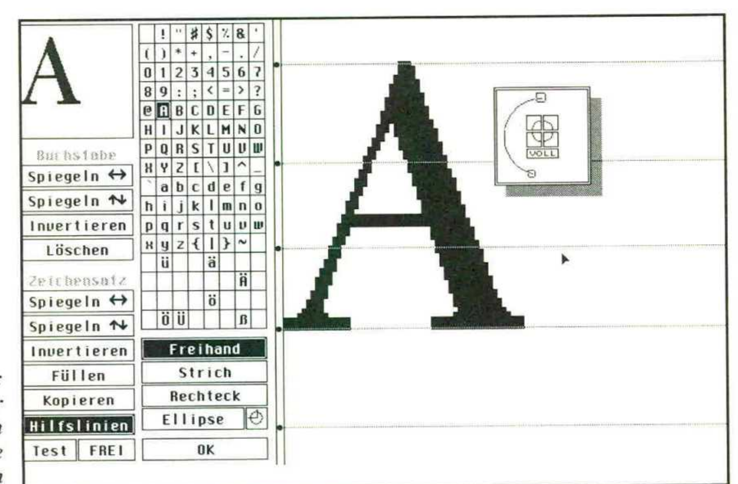


Bild 10: Sogar im Font-Editor lassen sich Kreisausschnitte zeichnen

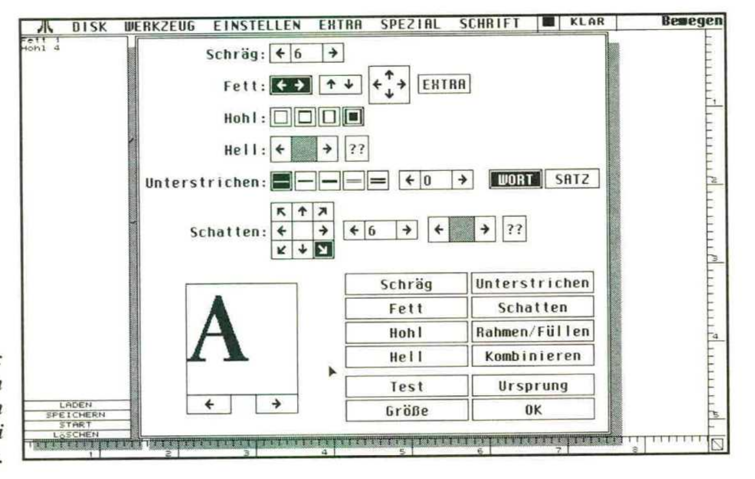


Bild 11: Die geladenen Zeichensätze lassen sich auf vielerlei Arten manipulieren.

den können. Durch die Programmfunktion Drehen kann ein rechtekkiger Ausschnitt gedreht und danach wieder in die Grafik eingefügt werden. Im Gegensatz zu manch anderem Grafikprogramm läßt sich der ausgeschnittene Bereich um alle drei Achsen drehen. Zwar kann beim Drehen zwischen Dichte minimal/maximal gewählt werden, die Routine bedarf aber noch der Verfeinerung, wie man bei Bild 6 (gedrehter Text) sehen kann. Der Menüpunkt *Spiegeln - Horizontal/Vertikal/Beides* erklärt sich von selbst. Der letzte Eintrag im Menü Spezial nennt sich Verfremden. Hiermit kann eine Grafik entweder fetter gemacht oder umrahmt werden. Vor allem das Umrahmen (bei Texten als "Outline" bekannt) ergibt mitunter recht schöne Effekte.

Eine große Auswahl von Zeichensätzen bietet das Programm CAG schon deshalb, weil sämtliche SIGNUM-Zeichensätze für 24-Nadel- (*.P24) und für Laserdrukker (*.L30) eingelesen werden können. Zusätzlich liest CAG natürlich sein eigenes, hausgemachtes Format (*.ZSA) und die Zeichen von Monostar Plus (*.SFT). Als letztes "Schmankerl" versteht die Software auch das "Headline"-Format (*.FHL). Wem die Zeichensatzauswahl immer noch nicht genügen sollte, der kann beim Hersteller gegen eine geringe Gebühr weitere Zeichensätze bestellen. Aber allein die mitgelieferten 16 sind recht schön gelungen. Mitunter sind zwar kleinere Höhenunterschiede bei normalerweise gleichgroßen Buchstaben vorhanden, die man mit dem Font-Editor jedoch in kürzerer Zeit ausgeglichen hat. Vielleicht paßt der Autor seine Zeichen ja noch selbst auf gleiche Zeichenhöhe an...

Im Zeichensatz-Editor (siehe Bild 10) kann der Benutzer seine Zeichen beliebig verändern. An Funktionen stehen ihm Spiegeln, Invertieren und Löschen einzelner Buchstaben oder des gesamten Zeichensatzes zur Verfügung. Ferner können einzelne Buchstaben kopiert werden, oder die gerade sichtbare Grafik läßt sich auf alle Buchstaben übertragen. Damit lassen sich einfache Zeichensätze entwerfen, bei denen alle Zeichen auf einer Grundform basieren. Zur besseren Übersicht bietet CAG auch fünf frei verschiebbare Hilfslinien an. Im Editor-Feld lassen sich mit der linken Maustaste in Schwarz, mit der rechten in Weiß entweder Freihandzeichnen, oder Linien ziehen und Rechtecke, Ellipsen oder Kreisausschnitte (nicht nur Vollkreise) zeichnen. Zwischendrin läßt sich durch den Knopf "Test" auch immer wieder das aktuelle Ergebnis des gesamten Zeichensatzes ansehen. Einzelne Buchstaben ausschneiden - etwa aus einer gescannten Grafikseite - und dann nachbearbeiten ist mit CAG ebenfalls kein Problem: Eine Art "Snapshot"-Funktion erlaubt es, Grafiken aus dem Arbeitsblatt zu "klauen", die in den Font-Editor übernommen werden. "UNDO" wurde im Zeichensatz-Editor ebenfalls nicht vergessen.

Besonders interessant gestalten sich die Manipulationsmöglichkeiten von Zeichensätzen (siehe Bild 11). Damit lassen sich die verschiedensten Effekte erzeugen. Neben einer einstellbaren Neigung des Zeichensatzes kann man diesen auch "verfetten", hohl machen, auf mehrerlei Arten unterstreichen und mit Schatten versehen. Ferner kann die Größe verändert werden. Die durchgeführten Veränderungen werden in einer Tabelle am linken Bildschirmrand angezeigt und zwischengespeichert. Diese läßt sich erneut aufrufen oder auf Datenträger abspeichern und erlaubt so die Anwendung auf beliebig viele Zeichensätze, ohne erneut alle Manipulationsarten neu anklicken zu müssen.

Das Schreiben mit den Grafikzeichensätzen kann auf zweierlei Arten erfolgen: entweder ein- oder mehrzeilig. Mittels einzeiliger Ausgabe kann - wie der Name schon sagt - maximal eine Zeile eingegeben werden, die in einem Mini-Editor sofort als Grafik sichtbar wird. Buchstaben- und Wortabstand lassen sich verändern, bevor man die Grafik in den Bildschirm einsetzt. Bei mehrzeiliger Ausgabe erscheint ein anderer Editor, in den sich Texte eingeben oder laden (ASCII-Format) lassen. Hier stehen Funktionen wie Blocksatz, Zentrieren usw. zur Verfügung. Ferner kann eingestellt werden, wie die Dehnung erfolgen soll - zwischen Worten oder Buchstaben. Danach kann der Benutzer in der Grafikseite einen viereckigen Bereich festlegen, in dem der

Text plaziert werden soll. Einziges Manko ist die fehlende Möglichkeit, Zeilen umzubrechen. Will der Benutzer etwa einen mehrzeiligen Text im Blocksatz ausgeben, wird es zur mühsamen Bastelei, bis alle Zeilen auf die maximal mögliche Länge eingetippt und umgebrochen sind. Aber bei CAG handelt es sich ja eher um ein Grafikprogramm als um eine Textverarbeitung...

Dateiformate

Es sollte ein Muß für jeden Handbuchautor sein, wenigstens die eigenen Dateiformate offenzulegen. Nur so ist eine Anpassung von Fremd-Software an die gekauften Programme möglich. Dieses Ziel wurde bei CAG optimal verwirklicht. Im Anhang fünf des Handbuches geht der Programmierer Byte für Byte auf die Dateiformate ein, die CAG erzeugt. Angefangen vom Arbeitsblatt (*.BLT) über die Bilder (*.DOO, *.ART, *.DA4), Objekte (*.OBJ) bis hin zu Zeichensätzen (*.ZSA, *.SFT) und Artenprogrammierung von Zeichensätzen (*.APR). Ferner werden erläutert: Objektverwaltung (*.OBV), die Stift- und Füllmusterdatei , die Vektorgrafikdatei und die abgespeicherten Programmparameter.

Fehlerbehandlung

Gut durchdacht ist das Kapitel sechs, in dem alle möglichen Fehler beschrieben werden, die auftauchen können. Hierzu hat der Programmierer die Fehler-Alert-Boxen mit Nummern versehen. Erscheint einmal eine solche Box, kann man unter der entsprechenden Nummer im Handbuch nachsehen und hat sofort die gewünschte Stelle gefunden, womit langes Suchen entfällt.

Hardware-Voraussetzungen

Für das Arbeiten mit CAG benötigt man einen ATARI ST mit mindestens ein Megabyte. Die normale Version braucht sogar zwei Megabyte. Für diejenigen, die aber nur einen "unaufgemotzten" 1040er oder ähnliches ihr eigen nennen können, wird auf der dritten Diskette eine "abgespeckte" Version für ein Megabyte mitgeliefert. Einige Programmteile (etwa das Drucken) wurden in ein zweites Programm ausgelagert. Das Programm CAG ist rein auf schwarzweiße Grafiken ausgelegt und arbeitet daher nur im hochauflösenden Modus mit einem SM124-Monitor (oder kompatiblen) zusammen.

Fazit

CAG ist ein Programm, das sich in erster Linie mit Pixelgrafiken beschäftigt. Ein Vektorteil ist zwar auch enthalten, dient aber eher der Erstellung von Pixelgrafiken und sollte auch als solcher angesehen werden. Nach kurzem Arbeiten mit "Computer Aided Graphics" merkt der Benutzer, daß viele "Goodies" enthalten sind, die manches möglich machen, was andere Grafikprogramme (noch) nicht können. Das Handbuch ist professionell gedruckt, und die sonst zu Hunderten üblichen Druckfehler vermißt man bei CAG gänzlich. Die Anleitung ist klar gegliedert und vermittelt die nötigen Informationen, ohne zu langweilen. Die Bedienung der Software ist einfach und äußerst übersichtlich. Das Konzept wurde überall gut durchdacht, was zum Beispiel bei den numerierten Fehlerboxen auffällt. An manchen Stellen im Programm wurden "??"-Knöpfe eingebaut, die etwa die Füllmustereinstellung erlauben, ohne den aktiven Menüpunkt dazu erst wieder verlassen zu müssen. Auch die im ganzen Programm beibehaltene Konsequenz (vor allem bei der Belegung der Maustasten) trägt zu einer schnellen Eingewöhnung bei. Bei vielen Funktionen lassen sich Parameter einstellen, die beim täglichen Zeichnen recht nützlich sind. Auch die Einbindung vieler Extras, wie komfortablem Zeichensatz-Editor, funktionsreicher Lupe, Drehen um die drei Raumachsen usw., stechen angenehm hervor. Auch die Offenlegung der Dateiformate für fortgeschrittene Anwender verdient einen Pluspunkt. Was eindeutig fehlt bei CAG sind große Negativpunkte. Die kleinen Fehlerchen, wie fehlende Lademöglichkeit von *.PAC-Grafiken, Grauverlauf nur in viereckigen Bereichen und ähnliches lassen sich angesichts der Funktionsvielfalt und der reichlich vorhandenen Extras jedoch sicher verschmerzen. Alles in allem ist CAG eine äußerst gelungene Software, die dem Anwender zu Erstellung von Pixelgrafiken nur wärmstens ans Herz gelegt werden kann. Der Preis von 349,- Mark ist zwar nicht gerade niedrig, erscheint für das Produkt jedoch angemessen.

RP

Bezugsquelle:

Stephan Stoske Software
Ludwigstraße 105
5600 Wuppertal
Preis: DM 349,-für registrierte Monostar-Besitzer DM 299,-Demodiskette DM 10,-

NEU & AKTUELL



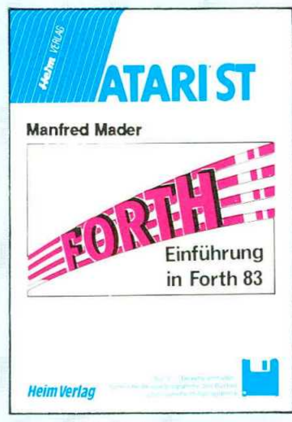
Buch incl. Programm-Diskette
Hardcover B-406
DM 59,-



Über 570 Seiten
mit Programmdisketten **DM 59,-**
Hardcover B-415



ca. 300 Seiten
Hardcover B-409 DM 49,–
Programmdiskette
zum Buch: D-249 DM 39,–



über 530 Seiten
Bestell-Nr. B-419 54,–
ISBN 3-923250-69-X
Inclusive Programmdiskette



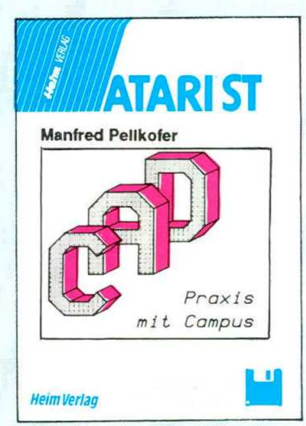
Über 300 Seiten
B-414
DM 49,-



Hardcover, über 430 Seiten
mit Programmdiskette
B-421
DM 69,-



Hardcover, 453 Seiten
mit Programmdiskette **DM 59,-**
B-400



ca. 290 Seiten
Bestell-Nr. B-418 59,–
ISBN 3-923250-67-3
Inclusive Programmdiskette



Hardcover
Bestell-Nr. B-436 **DM 59,-**
ISBN 3-923250-77-0
Inclusive Diskette mit Interaktiver Assembler-Entwicklungssoftware



220 Seiten – Hardcover
Bestell-Nr. B-432 **DM 49,-**
ISBN 3-923250-76-2
Inclusive Diskette mit Sicherheitssystem



Hardcover
Bestell-Nr. B-435 **DM 59,-**
ISBN 3-923250-79-7



über 330 Seiten
Bestell-Nr. B-420 54,–
ISBN 3-923250-70-3

Alle Preise sind unverbindlich empfohlene Verkaufspreise.

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir: ____________________

zuzügl. Versandkosten DM 6,– (Ausland DM 10,–) unabhängig von der bestellten Stückzahl

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

BASiCHART

Nicht nur eine *Tabelle*

Desk Datei Blatt Block Zelle Marke Output Chart Extra

AA0 ='Datei : FINANZ.WKF, Demo der Finanz-, Zeit- und Datums-Funktionen

	AA	AB	AC	AD	AE
0	Datei : FINANZ.WKF, Demo der Finanz-, Zeit- und Datums-Funktionen				
1	Bearb.:	T.Rubach	12.11.89	15:32	
2					
3	Eingabeparameter in Übergabereihenfolge...				Funktion
4	==========	==========	==========	==========	==========
5	Investition	Zinssatz	Perioden		AKTWERT
6	50.000,00DM	12%	20		373.472,18DM
7					
8	Investition	Zinssatz	Perioden		ZUKWERT
9	2.000,00DM	10%	20		114.550,00DM
10					
11	Kosten	Restwert	Lebensdauer	Zeitraum	AFADEG
12	10.000,00DM	1.200,00DM	8	1	2.500,00DM
13				2	1.875,00DM
14				3	1.406,25DM
15				4	1.054,69DM
16				5	791,02DM
17				6	593,26DM
18				7	444,95DM
19				8	134,84DM

Bild 1: Typisch Tabelle! BASiCHART kann seine Ähnlichkeit (zumindest im Bildaufbau) zu anderen "Tabellen" nicht verschweigen.

Quizfrage: Welche Gruppe von Anwenderprogrammen wird im privaten Bereich als allererste (in sinnvoller Absicht) genutzt?

Antworten:
a) Textverarbeitung,
b) Grafikprogramme,
c) Datenbanken,
d) Tabellenkalkulation.

Sie werden es nicht glauben! Es ist nicht die Textverarbeitung, denn wer hat zuhause so viel (privat) zu schreiben, daß es sich lohnt? Es ist ebenfalls nicht die Gruppe der Grafikprogramme, denn welche Hausfrau (bzw. welcher Hausmann) sieht Zeichnen oder Malen im Vordergrund häuslichen Schaffens? Nein, Datenbanken sind es auch nicht. So viele Adressen oder Kochrezepte muß niemand verwalten. Und nun die Lösung: JA! Es sind die Kalkulationsprogramme, die als erste (sinnvolle) Anwendung für hauswirtschaftliche Zwecke eingesetzt werden.

Irgendwie macht das auch Sinn. Denn mit einer Tabellenkalkulation läßt sich auf schnelle Art und Weise ein Haushaltsbuch führen oder eine Kontenverwaltung verwirklichen. Was natürlich in einem kleinen und überschaubaren Wirtschaftssystem wie dem Privathaushalt funktioniert, das muß doch für einen Handelsbetrieb gleichfalls gelten. Dies sollte natürlich nicht heißen, daß die Mehrzahl der Kalkulationsprogramme vornehmlich im privaten Umfeld zu finden sein werden.

Es kommt noch ein weiterer Aspekt hinzu: Die Mehrzahl der Programme wurde (leider) nicht für private Zwecke konzipiert. So finden wir die möglichsten und unmöglichsten Rechen-, Buchungs- oder Verwaltungsprogramme in allen Variationen. Die sind dann natürlich auf betriebliche Verhältnisse abgestimmt und kosten reichliches Geld. Das lohnt sich in keiner Weise für die "Minianwendung Privathaushalt".

Mit den Tabellenkalkulationsprogrammen ist aber eine Art "Anwenderoberfläche" geschaffen worden, die es (fast) unabhängig von der Größenordnung der Aufgabe erlaubt, alles (rechnerisch) zu lösen. Es soll auch hier nicht verschwiegen werden, daß es beim Umfang der Aufgabenstellung zwei Grenzen gibt:

1. die Geringfügigkeitsgrenze. Das bedeutet, daß es im Ausmaß der Rechenaufgabe eine Schwelle gibt, unter der die Benutzung der "Tabelle" sich einfach nicht lohnt. Denn wer nimmt die Einschalt- und Startprozedur auf sich, um einfach mal eine kleine Kolonne von Zahlen zu addieren? (Es sollen ja auch weiterhin noch Taschenrechner verkauft werden, bis Kopfrechnen wieder im Mode kommt.)

2. die Komplexitätsgrenze. Diese Schwelle wird dann übertreten, wenn die Tabelle zu groß geworden ist und Rechen, Sortier-, Formatier- oder Bewegungsvorgänge unvertretbar lange dauern. So wäre es sicher nicht empfehlenswert, eine komplette Industriebilanz oder eine Mandantenbuchhaltung per Tabelle zu erledigen. Dann machen die oben erwähnten Spezialprogramme wieder mehr Sinn.

Ganz abgesehen davon stellt der verwendbare RAM-Speicher ebenfalls eine Grenze nach oben dar. Man muß sich gleich hier von der Illusion lösen, die theoretisch möglichen 676 Spalten nebeneinander und die 10.000 Zeilen untereinander alle vollständig und randvoll benutzen zu können. Selbst wenn in diesen 6.760.000 Zellen jeweils nur 1 Byte abgelegt wäre, würde dies selbst den Mamumtspeicher eines MEGA-ST4 weit überschreiten.

Die "Rechen-Oberfläche"

Nach einer etwas mühsamen Installation (BASiCHART benutzt GDOS, und das muß angepaßt werden) meldet sich ein gewohntes Bild: Desktop-Leiste, Zeilennummern, Spaltenbuchstaben (wer hat noch nie eine Tabelle gesehen?). Besonders deutlich wird auch bei BASiCHART, daß sich bewährte Gepflogenheiten ähnlicher Programme (z.B. die Bildaufteilung, Edierzeile oben, bestimmte Bezeichnungen) einfach nicht verbessern lassen - das ist auch nicht weiter tragisch. BASiCHART sieht wirklich auf den ersten Blick kaum anders aus als VIP-Professional, LDW-Powercalc oder SciGraph (um nur einige zu nennen). Also muß es etwas (bzw. einiges) geben, worin sich BASiCHART von den Mitbewerbern unterscheidet. Sehen wir einfach mal weiter - Schritt für Schritt.

Dort wo sich eine Zeile und eine Spalte treffen, spricht man von einer Zelle. Die wird mit dem Cursor angesteuert, worauf dann in diese Zelle eine Eingabe gelegt werden kann. Soweit - sogut. Blöcke gibt es auch, das ist ein definierter Bereich zusammenhängender Zellen, zu dem Zweck, eine Aktion gemeinsam auf diesen Bereich wirken zu lassen. Beispiele: So sollen alle Zellen in einem Bereich in bestimmter Art und Weise formatiert sein, oder ein festgelegter Block soll an eine andere Stelle kopiert werden, oder es soll von den Zahlen innerhalb eines Blockes die Summe gebildet werden. Bewegen im Arbeitsblatt mit Maus oder Cursor-Tasten ist ebenfalls bekannt bewährt. Sehen wir weiter: Zellen edieren. Man kann eingeben:

1. Text bis 240 Zeichen Länge (zentriert, links- oder rechtsbündig)
2. Zahlen (d.h. konstante Werte)
3. Formeln (also Rechenvorschriften), wobei die Formel selbst in der Zelle nicht sichtbar wird, sondern nur ihr Ergebnis
4. Füllzeichen, die immer eine komplette Zelle ausfüllen, z.B. für Unterstreichungen. Sie passen sich in ihrer Anzahl immer der aktuellen Zellbreite an.
5. eine Folge von Hexadezimalzahlen zwischen den Grenzen 0 und FF jeweils getrennt mit einem Komma. BASiCHART erkennt diese Hex-Folge an dem Symbol "|" am Begin der Eingabe. Damit läßt sich eine Abfolge von Steuercodes (mit Kommentar) in Zellen legen, z.B. für eine Änderung der Druckereinstellung bzw. der Schriftattribute beim Ausdruck.
6. ein Block. Und das bedarf einer ausführlichen Erklärung. Normalerweise ist ein Block doch ein zusammenhängender Bereich von mehreren benachbarten Zellen. Wie kommt nun aber ein solcher Block in eine Zelle hinein? In Wirklichkeit wird nur die Definition eines Blocks (als Zeichenkette kodiert) in diese eine Zelle gelegt. So läßt sich sehr bequem auf bestimmte Blöcke verweisen oder diese in ihren Ausmaßen (Begrenzungen) manipulieren, z.B. um sie in Abhängigkeit einer Rechenoperation nachträglich neu zu begrenzen.

Diese beiden Funktionen, die ich bisher bei noch keiner anderen Tabellenkalkulation gefunden habe, erscheinen mir sehr nützlich. Neu ist außerdem, daß BASiCHART bei Formeln die Eingabe in "umgekehrter polnischer Notation" (UPN) zuläßt. Wer kennt die HP-Taschenrechner mit ENTER-Taste?

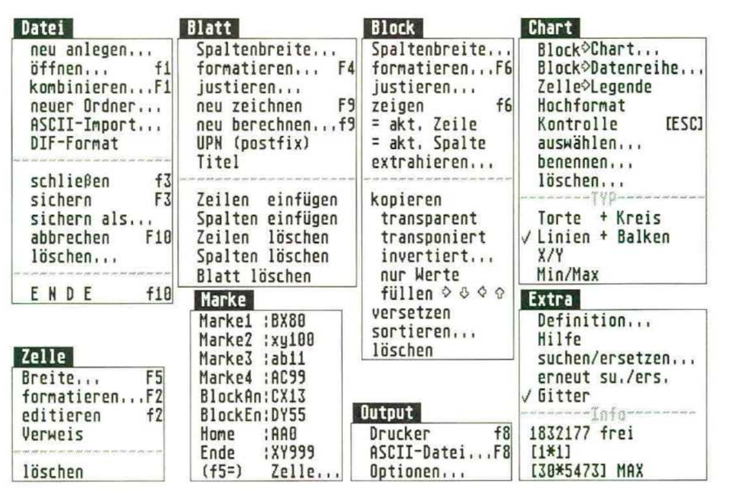


Bild 2: Ein Programm, das überwiegend auf Pull-Down-Menüs aufbaut, muß derer natürlich viele bieten.

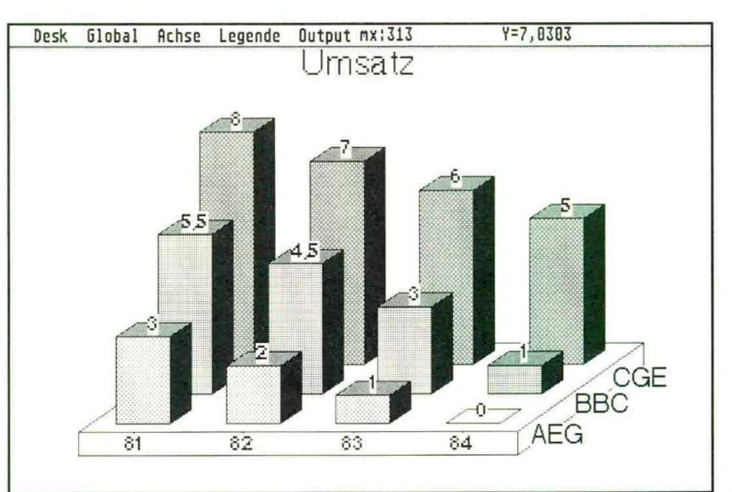


Bild 3: In einem angeschlossenen Grafikprogramm präsentieren sich die Zahlen höchst anschaulich.



Bild 4: So sehen die Menüs im Grafikteil aus. Hier kann auch die Taste als Alternative benutzt werden.

Ein kleines Rechengenie?

Gehen wir davon aus, daß Texte (z.B. Überschriften) sowie Zahlen eingegeben sind, dann stellt sich die Frage, welches Rechenwerkzeug haben wir zur Verfügung? Die Liste der Operatoren und Funktionen ist beeindruckend; um nur einige zu nennen:

1. arithmetische: Grundrechenarten, Potenzieren,
2. logische: Vergleiche, Negation, Implikation,
3. mathematische: Winkelfunktionen, Exponent, Logarithmen, Wurzel, Zufallszahl,
4. log. Verknüpfungen, Bedingungen,
5. statistische,
6. Datums- und Zeitrechnen,
7. Finanz: Abschreibung, Zins,
8. Vektor und Matrix,
9. UPN-Operationen,
10. String-Bearbeiter: Zeichen ersetzen, Codeumwandlung, Teilstringsuche,
11: Sonderfunktionen: Zellberechnung, Kombinationen, Blockoperationen.

BASiCAHRT beinhaltet nicht nur alle Rechenvorgänge der "Mutter aller Tabel-

len" LOTUS 1-2-3 (Version 2.0), sondern kommt auf die stattliche Zahl von 130 verschiedenen Funktionen. BASiCHART rechnet auf 14 Stellen genau und hat einen Zahlenbereich bis zu Exponenten von +/- 300.

Ein weiteres Bonbon, das ich von anderen Programmen her nicht kenne: Es ist möglich, zusätzlich noch weitere 100 Konstanten, Zellbereiche oder Rechenfunktionen, mit bis zu 11 Argumenten, selbst zu definieren und darin auch die schon bestehenden (siehe oben) mitzuverwenden. Man kann sich quasi eigene Phantasieformeln zusammenbasteln.

Bild 5: Selbst für einzelne Blöcke der Tabelle sind ganz individuelle Ausgabeparameter möglich.

Bild 6: Im Hilfemodus erscheint diese unübersichtliche Funktionsauswahl, die dann zur weiteren Begriffsklärung eine manuelle Eingabe erwartet.

Zahlen anschaulich gemacht

Wie schon oft erwähnt, macht es wenig Sinn, ellenlange und öde Zahlenkolonnen einem breiteren Publikum vorzulegen, weil sie sehr unübersichtlich und wenig aussagekräftig sind. Gerade bei Kalkulationen (im wahrsten Sinne des Wortes) geht bekanntlich Probieren über Studieren. Beim Testen bestimmter Zusammenhänge wird es nötig, dort einen Wert zu ändern, irgendwo eine Formel umzustellen und dann die Auswirkung (oft auch im Zusammenhang mit anderen Ergebnissen) festzustellen. Dann erweist sich die grafische Darstellung in einigen Kurven oder Balken als besseres Mittel, Veränderungen zu überblicken.

BASiCHART verfügt über einen reichen Schatz an grafischen Instrumenten. Die Grafikauswertung ist in einem eigenen Programmteil untergebracht. Man gelangt dorthin, wenn z.B. eine Zahlenkolonne als Block definiert und per Menü "CHART" die Art der Übernahme angewählt ist, d.h. eine Übernahme der Zahlen aus dem Block als grafische Signale oder in eine Datenreihe sowie die Übernahme einzelner Zellen als Legende.

Alle Grafikteile sind als eigenständige Objekte deklariert. Das bedeutet, daß jedes Grafikobjekt individuell manipuliert werden kann, z.B. kann man es vergrößern/verkleinern, verschieben, dehnen, stauchen usw. Die Bildinhalte lassen sich von Hand auch nachträglich verändern.

Die Auswahl der Darstellung kann sich sehen lassen:

- Linien
- Treppen
- Balken
- 3D-Balken
- Flächen
- Kreis
- Torten
- X/Y
- Min/Max

So passen beispielsweise in ein Chart-Bild bis zu 3 Diagramme, auch das Mischen oder Kombinieren verschiedener Darstellungsarten zu einem Bild ist möglich. Nahezu alle Grafikattribute (z.B. Schriftgröße und -stil, Füll- und Hintergrundmuster, Farbe, Transparenz) sind manuell veränderbar. Bis zu 10 Charts lassen sich pro Rechenblatt erstellen, zwischen denen einfach per Knopfdruck hin- und hergeschaltet werden kann.

Ein Superbonbon: Es lassen sich einzelne Zellen in das Grafikfenster hineinlegen. Damit sind bei Werteänderung in diesen Zellen die Auswirkungen in der Grafik unmittelbar beobachtbar. Das habe ich auch bei keinem anderen Programm gefunden. Dies muß auch nicht unbedingt bedeuten, daß diese Zellen dann zur Grafik gehören (z.B. bei einem Ausdruck sichtbar), es ist gewissermaßen nur eine Projektion von Teilen des Arbeitsblattes in das Grafikfenster.

Es bedarf fast nicht mehr der Erwähnung, daß Achsen auch logarithmisch sein können, daß die Grenzen des Darstellungsbereiches für die Achsenwerte manuell oder automatisch ermittelnd einstellbar wären, eine 3D-Tiefe (Anteil des Perspektivraumes am Gesamtbild) wählbar, die Position des Mauszeigers mit seinen Koordinaten ablesbar ist usw.

Was man Schwarz auf Weiß hat

Wenn dann die Arbeit auf dem Bildschirm ein befriedigendes Ergebnis zeitigte, möchte man doch gerne das Ganze auf Papier festhalten. Für das Rechenblatt wird auf 1st_Word-Druckertreiber zurückgegriffen, so daß alle gängigen Drukker ansprechbar sind. BASiCHART schreibt in Dateien im ASCII- und DIF-Format und liest ASCII, DIF, CSV und WKS, dies gilt ebenfalls für das Rechenblatt.

Wie schon erwähnt, greift BASiCHART für die Grafik auf den Gerätetreiber GDOS zurück und benutzt auch dessen Fonts. An Grafikformaten versteht BASiCHART (für Pixel) PIC, Degas-PI?, IMG, (für Vektor) Easydraw-GEM und Metafile-GEM.

Dies erlaubt die professionelle Weiterverarbeitung von Grafik in höchster Auflösung für DTP-Programme wie CALAMUS.

Bild 7: Viele Funktionen bringt BASiCHART schon mit. Neue, unbekannte, erweiterte oder (vom Benutzer) zusammengestelle Definitionen erweitern den Funktionsvorrat ins Unermeßliche.

Bild 8: Für jede einzelne Zelle kann aus einem großen Vorrat die Formatierung ganz speziell eingestellt werden.

Was an BASiCHART auffiel

Es sind wieder einmal die Kleinigkeiten, die das Besondere eines Programms ausmachen. Feinheiten, wie das nachträgliche Definieren komplizierter Formeln, Eingabe im Hex-Format oder Projizieren einzelner Zellen in das Grafikfenster, die sind es, die den Reiz dieses Programms ausmachten. Ich will ehrlich sein: Zu Beginn des Tests dachte ich, schon wieder eines dieser unzähligen Abwandlungen von LOTUS für den ATARI ST. Erst nach und nach bemerkte ich die kleinen Besonderheiten, die man in keinem ähnlichen Programm findet. Man muß halt mit BASiCHART gearbeitet haben, um diese Besonderheiten zu erkennen und wertmäßig einzuordnen. Man spürt an vielen Stellen, daß ein Entwickler an diesem Programm arbeitete, der die Mängel früherer LOTUS-/ATARI ST-Derivate kannte. Zudem hat der Rechenteil von BASiCHART einen längeren Reifeprozess hinter sich. Wenngleich der CALC-Teil (Rechenblatt) von BASiCHART uneingeschränkt die Note 1 verdient hat, gibt es für den CHART-Teil (Grafikfenster) geringfügige Verbesserungsvorschläge. Alles in allem ist BASiCHART ein sinnvolles, gut durchdachtes Programm.

Ein weiterer Pluspunkt in diesem Gesamtbild ist das Handbuch. Es ist vorbildlich gegliedert: Zuerst ein Einführungsteil mit wichtigen Vereinbarungen, ein Kapitel zum Schnelleinstieg (für ungeduldige Durchstarter), dann folgt ein Kapitel "Schritt für Schritt" (für den geduldigen Anwender) und ein Kapitel, das Menüpunkt für Menüpunkt haarklein erklärt. Diese Struktur gilt für den Kalkulations-, als auch für den daran anschließenden Grafikteil. Ein kleiner Anhang bringt zwei Anwendungsbeispiele: 1. lineare Regression (warum nur so etwas Kompliziertes?) und 2. eine Umsatzaufstellung.

Die Gestaltung ist ansprechend und mit einer ausreichenden Anzahl Bilder aufgelockert.

Zu guter Letzt: der Preis. BASiCHART kostet 198,- DM. Wer auf die Grafik verzichten möchte, erhält BASiCALC zum Preis von 98,- DM.

DK

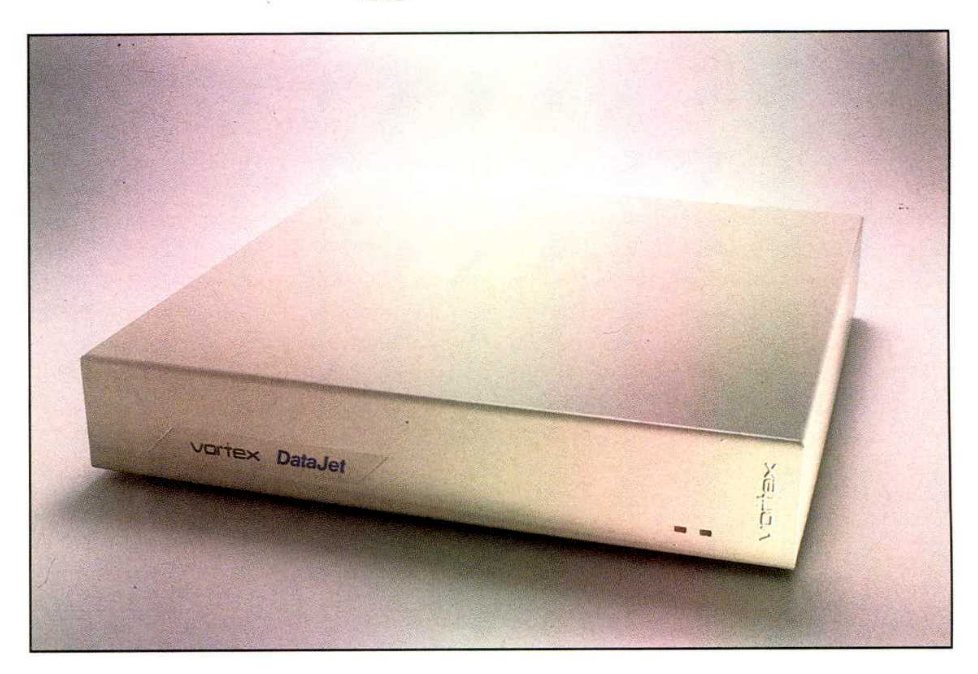
Bezugsquelle:

POINT Computer GmbH
Gollierstraße 70/C5
8000 München 2
Tel.: 089/505657





Ablösung



Vortex Datajet X60

Bisher kannte man Vortex auf dem ST-Markt nur durch die HDplus-Plattenreihe. Nun diversifizieren die Fleiner: Neben einer Ethernet-Lösung bietet Vortex einen AT-Emulator an, und die HDplus-Festplattenreihe wird durch die DataJet-Platten abgelöst. Ein Exemplar dieser Reihe, die DataJet X60, haben wir im Test hochnotpeinlich befragt.

Vortex verfolgt bei der DataJet-Reihe - endlich - ein neues Konzept: Statt der nur noch bei Original-ATARI-Platten und PCs üblichen ST506-Laufwerke werden SCSI-Laufwerke mit integriertem Controller verwendet, die nicht nur neue Möglichkeiten bieten, sondern gemeinhin auch wesentlich schneller sind. Dazu trägt in unserem Falle besonders bei, daß die Vortex-Platten sich nun auch - wie fast alle anderen Platten für den ST - mit Interleave 1 formatieren lassen (bei der HDplus-Reihe meistens Interleave 3). Was das bedeutet? Das heißt, daß die Sektoren auf einer Spur innerhalb einer Umdrehung des Plattenstapels eingelesen werden können (HDplus: 3 Umdrehungen) - was die Transfer-Rate gegenüber HDplus-Platten glatt verdreifacht.

Aufsteiger

Die mittleren Zugriffszeiten liegen bei der DataJet-Reihe laut Vortex durchweg bei etwa 24ms, auch das eine Steigerung. Fragt sich der Laie und der Fachmann wundert sich: Was steckt denn nun in Vortex' Datendüser?

In der appetitlichen, aber für Jet-Verhältnisse nicht besonders aerodynamischen Verpackung - ein stabiles Metallgehäuse, grau in grau mit abgerundeten Ecken - steckt das Vortex-Microboard, ein Netzteil, ein Lüfterchen und das Laufwerk, in diesem Falle ein SCSI-Laufwerk von Seagate (ST177N). Übrigens faßt dieses nur etwa 58MB und nicht 60MB, zumindest wenn man 1MB zu 1024 Kilobyte und 1KB zu 1024 Bytes ansetzt - wie es sich gehört.

Intelligenz inklusive

Das Microboard ist eine Vortex-Eigenheit: Es umfaßt nicht nur einen Host-Adapter zum Anschluß an den DMA-Bus des ST, sondern einen kompletten kleinen Z80-Rechner, der für einige Extra-Gimmicks zuständig ist. Dieser Z80-Rechner leitet SCSI-Kommandos an das SCSI-Laufwerk weiter und fängt spezielle Vortex-Kommandos ab, um sie auszuführen. Beispielsweise lassen sich einzelne Partitionen per Hardware schreibschützen - ein zuverlässiger Schutz gegen Rechnerblindflüge kleiner Geschwister, Intrigen übelwollender Kollegen oder Attacken infektiösen Rechnergetiers. (Merke: Es heißt "das Virus".)

Außerdem kann das Microboard den Motor der Platte nach einer voreingestellten Zeit oder auf Kommando des Rechners ausschalten, das Rauschen des Lüfters je nach Temperatur regeln, die Köpfe des Laufwerks rückwärts einparken und bei der Diagnose von Fehlern helfen. All das sind recht ungewöhnliche Zusatzlei-

stungen einer Platte. Den Motor der Platte per Tastendruck auszuschalten, um mehr Ruhe auf dem Schreibtisch zu haben, halte ich allerdings für keine besonders plattenschonende Angelegenheit; Vortex meint dazu, zwar nutze der Motor durch das Aus- und Anschalten etwas ab, dafür aber schone man die Lager durch die Ruhepausen. Gehen Sie am besten sparsam mit dieser Möglichkeit um; das empfiehlt übrigens auch Vortex.

Damit ist das Microboard noch nicht am Ende. Oft hängen sich herkömmliche Host-Adapter auf, wenn beispielsweise der Rechner im falschen Moment - etwa beim Hochlaufen der Platte - eingeschaltet wird, oder wenn der Rechner abstürzt oder zwei Platten die gleiche Target-Nummer haben. Solche Fälle fängt das Microboard ab, indem es den Verkehr auf dem DMA-Bus selbst auf Plausibilität überwacht. Sie können den DataJet auch zusammen mit dem Rechner einschalten; während die Platte hochläuft, liefert das Microboard der Boot-Routine des ST einen gefälschten Konfigurationssektor, der ein Programm enthält, das wiederum wartet, bis der DataJet tatsächlich hochgelaufen ist.

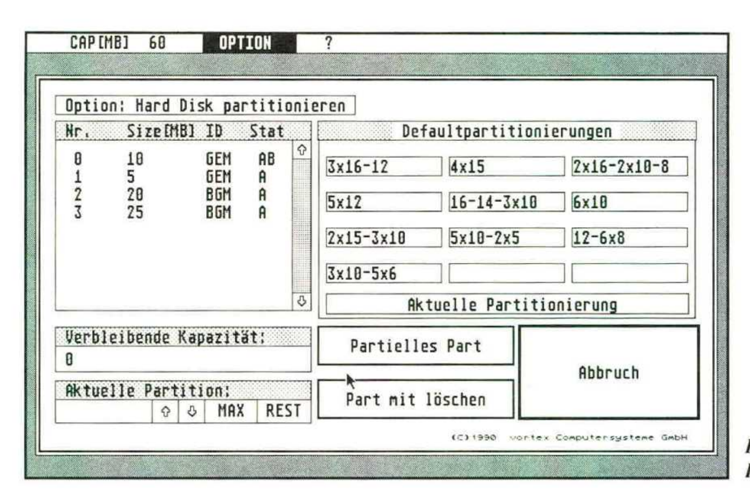


Bild 1: Der Partitionierdialog

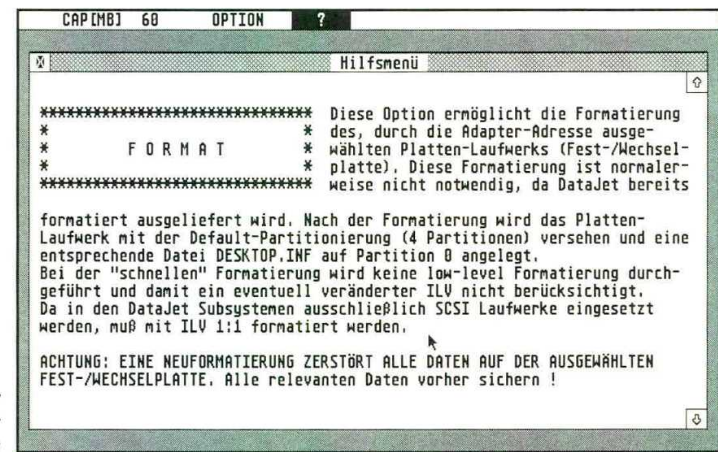


Bild 2: Der Plattenklempner gibt Hilfestellung

Das Microboard unterstützt seit neuestem auch beliebige SCSI-Kommandoklassen. Zur Erläuterung: Das ACSI-Protokoll auf dem DMA-Bus des ST schränkt den Befehlsumfang von SCSI-Geräten ein; das kann man mit dem Microboard umgehen. Wie man das macht, steht in einem "Service- und Programmierhandbuch", das Vortex allerdings nur auf Anfrage und gegen Kostenerstattung und bevorzugt an Händler ausgibt.

Flüsternder Jumbo

Glücklicherweise ist der (!) DataJet X60 wesentlich leiser als seine schnittigeren fliegenden Namensvettern. Sowohl Lüfter als auch Laufwerk geben nur ein zufriedenes Räuscheln von sich: Prädikat "angenehm". Vortex hat sich in diesem Punkt gar um offizielle Zustimmung bemüht: Für den DataJet X40 hat man sich vom TÜV Südwest einen maximalen Geräuschpegel von 34dBA bescheinigen lassen, Zitat aus dem Gutachten: "Die Werte der VDI 2058 Blatt 3 und der Arbeitsstättenverordnung von 55dBA für Lärm am Arbeitsplatz bei geistiger Tätigkeit wurden deutlich unterschritten. Das Arbeitsgeräusch eines ATARI ST1040-Computers (Tastenbetätigung ohne Keyclick) ist mit 55dBA deutlich lauter."

Wäre nur anzumerken, daß Tastenklicks kein Dauergeräusch und daher nicht unbedingt ein fairer Maßstab sind. Aber was soll die Haarspalterei, die Platte ist in punkto Lautstärke business class.

Neues Prozessorfutter

Zur neuen Platte gibt es auch neue Software. Sie ist zum Teil nur für die DataJets gedacht und nicht mehr mit den HDplus-Platten lauffähig, wie Vortex auch anmerkt. Da wäre zunächst der Treiber, der auch die neuen Partitionentypen unterstützt, die ATARI mit AHDI 3.0 eingeführt hat. Die maximale Partitionengröße liegt nun bei 256 Megabytes - und nicht bei 512MB, wie ich irrtümlich in den Plattentests der April-Ausgabe verbreitet habe ('tschuldigung!). Vortex erlaubt zwar beim Partitionieren 512MB, aber davon sollten Sie die Finger lassen. Während ATARI theoretisch beliebig viele Partitionen zuläßt, wird in der Vortex-Gebrauchsanweisung nur von 12 Partitionen pro Laufwerk gesprochen. Tatsächlich kann der Treiber aber mehr Partitionen erkennen und der Installateur bis zu 32 Partitionen anlegen.

Übrigens läßt sich der DataJet auch mit anderer gängiger Platten-Software bearbeiten; partitioniert man ihn mit anderer Software, versteht das wiederum auch der Vortex-Treiber. Nur der Vortex-Partitionierer nimmt solches Treiben leicht übel, da er die gültigen Partitionierungsmöglichkeiten, wie sie ATARI vorsieht, unverständlicherweise einschränkt und darum nicht alle AHDI3-Partitionierungen versteht. Meldungen wie "Kann Konfigurationssektor nicht finden" oder "Hardware-Schreibschutz aktiviert" an völlig unerwarteten Stellen sind die Folge, und zuweilen trat das Installationsprogramm kommentarlos aus. Nach ein paar hartnäkkigen Versuchen allerdings erledigte sich das Problem meist von alleine.

Läßt man ab von derlei sündigem Tun, verhält sich auch der Plattenklempner aus dem Hause Vortex friedlich (Bild 1). Er bietet alle gängigen Operationen wie Formatieren, Partitionieren, Installieren eines Treibers und das Einstellen von Treiberparametern. Praktisch ist die Hilfestellung im Programm (Bild 2).

Weitere nützliche Beigaben: Ein Parkprogramm, ein Accessory zur Kontrolle der wichtigsten Treiberparameter (Auslastung des Caches, Schreibschutz für einzelne Partitionen et cetera, siehe Bild 3) und zwei Backup-Programme namens BACKPART und BACKDISK. BACKPART kopiert von Partition zu Partition, was sehr schnell geht, allerdings nur unter ein paar Vorbedingungen funktioniert: Die Zielpartition muß mindestens so groß wie die Quellpartition sein; sie muß aber auch mit der gleichen Sektorgröße arbeiten. Das Kopieren von einer GEM-Parti-

	AHS-Q105	SH205/40MB (ST251/1)	DataJet X60	DataJet X60 (mit CHECKHD8)
minimale Zugriffszeit				
offiziell	6ms	8ms	8ms	8ms
gemessen	5.2ms	7ms	6ms	4.6ms
mittlere Zugriffszeit				
offiziell	19ms	28ms	24ms	24ms
gemessen	26ms	31.7ms	26.3ms	23.3ms
maximale Zugriffszeit				
offiziell	40ms	70ms	44ms	44ms
gemessen	35ms	61ms	39.9ms	36.6ms
max. Transfer-Rate				
ohne Zylinder-wechsel	860KB/s	510KB/s	645KB/s	645KB/s
mit Zylinder-wechsel	756KB/s	436KB/s	774KB/s	774KB/s

(Tabelle 1: hardwarenahe Meßwerte)

tion (<16MB) auf eine BGM-Partition (>16MB) ist damit also nicht möglich. Abgesehen davon ist dieses Programm höchst nützlich und zeitsparend, ganz besonders, weil es den Trend zur Zweitplatte enthält.

BACKDISK dient, o Überraschung, der Datensicherung auf Diskette. Es arbeitet nicht dateien-, sondern sektorweise. Das hat zwar zur Folge, daß Sie unter Umständen wesentlich mehr Disketten verwenden, andererseits vereinfacht es die Handhabung und ermuntert so vielleicht eher, die unangenehmen Sicherungsarbeiten von Zeit zu Zeit auf sich zu nehmen. Leider kann auch BACKDISK noch nicht mit HD-Disketten umgehen [1]; der Versuch, dem Programm die passenden Diskettenparameter beizubringen, wird mit Abstürzen nicht unter zwei Bömbchen bestraft. Zudem kann man mit BACKDISK nur auf Laufwerk A: sichern - eine Einschränkung, die ich nicht verstehe. Die entstehenden Sicherungsdisketten sind übrigens "nicht GEMDOS-kompatibel", man kann von ihnen kein Verzeichnis lesen.

Der Datensicherheit dient ein Helferchen, das auf einer Platte, auf der der Konfigurationssektor gelöscht wurde, die Partitionen wiederfindet und den Konfigurationssektor wiederherstellt - leider nur für Vortex-Platten gedacht. SHOWBAD erstellt aus einer Datei, die der Platteninstallateur beim Prüfen der Platte erstellt, einen mit ASCII-Editoren lesbaren Text, der die aufgetretenen Fehler genau lokalisiert.

Kompabeatle oder nicht?

Der Plattentreiber kann inzwischen auch von allen Partitionen booten (gleiche Methode wie in CBHD); das gilt auch für Accessories. Natürlich sorgt der Treiber auch für die Linderung des 40-Ordner-Problems, und zu allem Überfluß bietet er uns einen integrierten Cache, den man in der Größe frei einstellen kann. Erwarten Sie unter TOS 1.4 aber von einem Cache des verwendeten Typus keine Wunderdinge - die GEMDOS-eigenen Sektorpuffer aufzustocken, lohnt sich wohl eher.

Diese Sektorpuffer lassen sich mit der Vortex-Software leider nicht erweitern. Wenn man es vertrauensselig mit dem ATARI-offiziellen Cache-Erweiterungsprogramm oder mit seinem Pendant aus dem Scheibenkleister versucht, die sich auf die AHDI-Kompatibilität des verwendeten Treibers verlassen, bleibt der Rechner hängen, sobald man auf Partitionen mit mehr als 16 MB Umfang zugreift. Kompabeatle zu AHDI V3.x ist die Vortex-Software also nicht ganz. Bei Vortex hieß es dazu, man habe dies auch nie beabsichtigt; beispielsweise laufe die Medienwechselerkennung bei Wechselplatten sowieso ganz anders als beim ATARI-Vorbild. Für Menschen, die immer noch mit TOS 1.0 oder TOS 1.2 arbeiten, gibt es eine gute Nachricht: Der Treiber installiert wahlweise einen Tastatur-Reset, was Ihnen die rituellen Verneigungen vor Ihrem Mega ST erspart.

Die HDplus-Reihe zeichnete sich ja nicht gerade durch vorbildliche Kompatibilität aus; wir erinnern hier an gewisse Schwierigkeiten mit dem MODE-SENSE-Kommando. Vortex hat daraus gelernt und die speziellen Vortex- von den Standard-SCSI-Kommandos deutlich unterschieden; alle SCSI-Kommandos des jeweiligen eingebauten Laufwerks werden unterstützt. Kollisionen mit Anwenderprogrammen sind nun wohl nicht mehr zu befürchten. Alle Vortex-eigenen Parameter werden sauber in einer eigenen speziellen Partition vom Typ SYS geführt, die schreibgeschützt ist; der früher übliche interne Systembereich wird im wesentlichen dadurch abgelöst.

Cache-Größe in Sektoren:	0	128	256	512	1024	Werte für AHS105Q
100*50 Sektoren lesen	5	5	5.1	5	5	n/a
100*50 Sektoren einzeln	86.6	7	6.3	7	7	n/a
Dateien anlegen	10.4	11.1	11.2/ 9.7(*)	11/ 9.7(*)	11/ 9.7(*)	10.3
Dateien lesen	1.7	1.1	1.1	1.1	1.1	1.1
Dateien löschen	2.75	2.5	2.5	2.5	2.5	3

(*) beim zweiten Versuch

Tabelle 2: BIOS- und GEMDOS-nahe Meßwerte in Sekunden

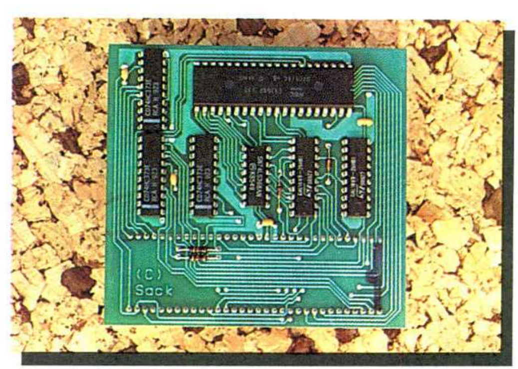
Ein echter Überflieger?

Da wir gerade nicht davon sprechen: Die üblichen Torturen - gemeinhin unter dem euphemistischen Begriff "Benchmark-Paket der ST-Computer" zusammengefaßt - brachte der Delinquent ohne Klagen hinter sich; die Ergebnisse des Verhörs durch das hardwarenahe CHECKHD sind in Tabelle 1 zusammengefaßt. Wir ersehen daraus, daß die Hardware-Daten, insbesondere die Transfer-Raten, sich gegenüber der HDplus-Reihe erheblich verbessert haben und nun mit anderen Platten für den ST leicht mithalten können.

Die *MS-DOS*-Welt erschließen mit PC-Speed ...

PC Speed für den ST

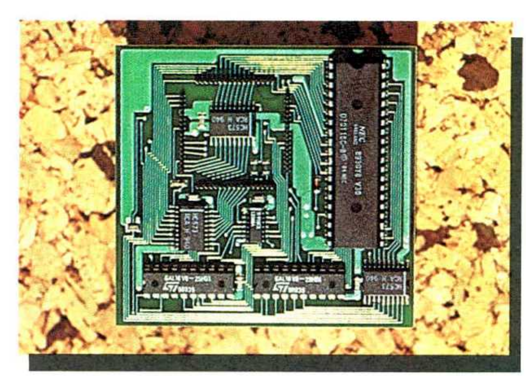
Über 20 000 Anwender haben weltweit in den ersten neun Monaten nach seiner Einführung PC-Speed gekauft und sich mit diesem Spitzenprodukt Zugang zu den Tausenden von MS-DOS Programmen verschafft.



PC-Speed erweitert den ATARI ST zum MS-DOS-Rechner

PC Speed für den STE

Obwohl PC-Speed für den STE erst seit vier Wochen auf dem Markt ist, haben schon über 1 000 Benutzer zugegriffen und integrieren MS-DOS Programme wie selbstverständlich in ihre Computerarbeit.



PC-Speed erweitert den ATARI STE zum MS-DOS-Rechner

Was hat PC-Speed diesen Erfolg gebracht?

- die modernste Technologie
- die saubere Lösung
- die sichere Funktion
- die hohe Leistung
- die HighTech-Qualität
- die Verfügbarkeit im Bundesgebiet bei über 400 Fachhändlern
- die gute Hotline-Betreuung
- der Update-Service

Mit PC-Speed profitieren ST- und STE-Anwender von der Leistung ihres Atari-Computers und gleichzeitig von der unübertroffenen Programm-Vielfalt der MS-DOS Welt

...die Überraschung

AT Speed



Die Erfahrung des Entwicklers H. J. Sack macht es möglich!

* MS-DOS ist eingetragenes Warenzeichen der Microsoft Corp.

Vertrieb weltweit:

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Bundesrepublik:
bei über 400 ATARI-Händlern

Österreich:
Darius
Inh. K. Hebein
Hartlebengasse 1-17/55
A-1220 Wien
Tel.: 00 43 - 222 - 239580

Schweiz:
Data Trade AG
Landstr. 1
CH-5415 Rieden-Baden

für alle anderen Länder:
COMPO SOFTWARE GmbH
Postfach 1051
D-5540 Prüm (FRG)
Tel.: 0 65 51 / 62 66

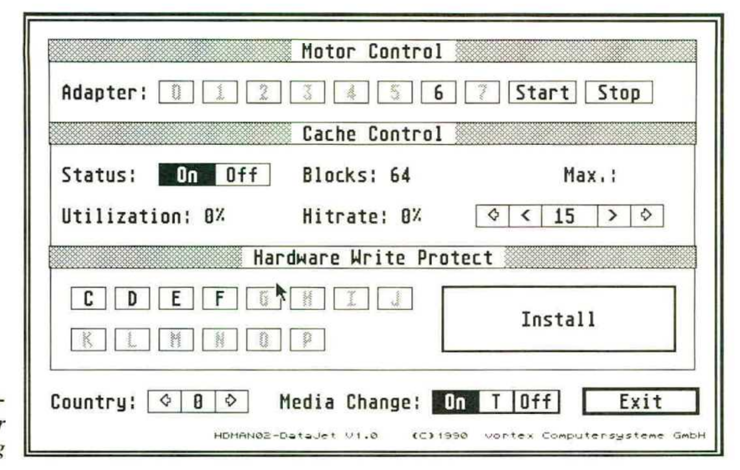


Bild 3: Das Hilfs-Accessory zur Treibersteuerung

Zur Tabelle 1 sind einige Anmerkungen am Platze: CHECKHD gibt es inzwischen in einer leicht überarbeiteten Version, die Eigenheiten von Seagate-SCSI-Platten berücksichtigt. Leider mußte ich dazu den Algorithmus der Spurwechseltests ändern - daraus ergeben sich andere Meßwerte als in der früheren Version. Dafür kann man die Meßwerte nun mit den offiziellen Angaben vergleichen, da die Meßmethode mit denen der Hersteller größtenteils identisch sein dürfte. Wie sich die neuen Algorithmen beim DataJet auf die Meßwerte auswirken, sehen Sie in der entsprechenden Spalte in Tabelle 1. In dieser neuen Version wird CHECKHD übrigens auf den Systemdisketten der Firma Vortex gleich mitgeliefert.

Tabelle 2 offeriert Ihnen einen weiteren Wust von Meßdaten: Hier geht es um Operationen, wie sie im täglichen Umgang mit einer Platte vorkommen. Dazu habe ich das dafür zuständige Programm HDBENCH auf eine 10-MB-Partition des DataJet losgelassen. Dieser Tabelle ist auch zu entnehmen, daß sich ein Cache mit mehr als 128 Blöcken nicht lohnt. Der Vergleich mit einer AHS105Q von FSE zeigt, daß die dort verwendeten Quantum-Laufwerke beim Transfer von Dateien die Nase vorn haben; Anwendungen, bei denen viele Kopfbewegungen nötig sind (Dateien anlegen, Dateien löschen), laufen nach diesem Test auf beiden Platten

Technische Daten:

Hardware	
Gehäuse:	Metall
Host-Adapter:	Vortex SCSI-Microboard
DMA-Bus	
durchgeschleift:	ja
gepuffert:	ja
SCSI-Anschluß:	nein
Target-Adresse	
Controller:	integriert
Aufzeichnung:	RLL
Laufwerk:	Seagate ST177N
Kapazität:	58MB
Köpfe:	5
Zylinder:	921
Sektoren/Track:	26
Sektoren:	118809
Autopark:	ja
Hardwarecache:	nein
Benchmarks	
CHECKHD	siehe Tabelle 1
HDBENCH	siehe Tabelle 2
Transfer via Treiber	619KB/s
mit Positionierung	545KB/s
Geräuschentwicklung:	
Laufgeräusch:	niedrig
Lüftergeräusch:	niedrig
Stepgeräusch:	niedrig
Software	
Kopierschutz:	nein
Treiber...	
autobootfähig	ja
AHDI3.0-komp.	ja (*)
erkennt Wechsel	ja
Partitionen	bel. (max. 14 unter GEMDOS)
max. Partitionsgröße	256MB
Root-Sektorformat	AHDI 3 (*)
Booten von ACCs	von bel. Part.
Booten v. AUTO-Prgs	von bel. Part.
Ordnererweiterung	im Treiber
wird abgelegt in	SYS-Partition
Schreibschutz	ja, per Hardware
Paßwortfunktion	nein
Verify	nein
Installation...	
Bedienung	sehr einfach
Konfigurierbarkeit	-
Defektprüfung	ja, auf Controller- und GEMDOS-Ebene
sonstige Software	BACKDISK, BACKPART, SCAN (rettet Konfigurationssektoren), Hilfs-Accessory, weitere kleinere Hilfsprogramme
Handbuch:	
Umfang	vollständig
Stil	verständlich
Programmierhinweise	keine
Besonderheiten:	Intelligenter Microboard-Host-Adapter: Hardware-Schreibschutz für einzelne Partitionen, alle SCSI-Kommandogruppen erreichbar, Intervall-Autopark, Motorabschaltung, Lüftersteuerung, Defektbehandlung, Erkennung von Fehlerzuständen auf dem DMA-Bus, Einschalten der Platte mit dem Rechner möglich
Kompatibilität	
Kompatibilitätsklasse	AHDI-kompatibel
Bekannte Inkompatibilitäten:	Cache-Programme für den GEMDOS-eigenen Sektorpuffer, evtl. auch andere Cache-Programme
arbeitet mit SLM804	ja
Preis:	DM 1998.-

(*) mit kleinen Einschränkungen, siehe Artikel

etwa gleich schnell. Der Vollständigkeit halber sollte ich noch erwähnen, daß das Anlegen von Dateien durch den Cache-Speicher bisweilen schneller wurde, wenn man den HDBENCH-Test zweimal hintereinander auf die gleiche Partition losließ; alltagstypisch sind die erzielten Verbesserungen wohl nicht. Wie man weiterhin sieht, kann ein Cache-Speicher herkömmlicher Bauart zumindest unter TOS 1.4 bestimmte Operationen (hier beim Test "Dateien anlegen") sogar leicht verlangsamen.

Bookware

Die Gebrauchsanweisung, gut 80 Seiten dick, ist wie die Software einfach zu verstehen und weist auf viele mögliche Bedienungsfehler hin. Auch auf den theoretischen Hintergrund wird Wert gelegt, ohne daß man sich in Details verliert. Bilder assistieren, wo es nötig ist, wenn sie auch an einigen wenigen Stellen etwas unglücklich verteilt sind. Eine alles in allem angenehme Gebrauchsanleitung mit nur wenigen Unklarheiten. Leider fehlen jegliche Programmierhinweise - dazu muß man auf das Vortex-Service-Handbuch zurückgreifen.

Lobpreis und Schelte

In der DataJet-Reihe sind Platten von 30 bis 180MB Kapazität mit Laufwerken von Seagate und Fujitsu erschienen, auch Kombinationen mit einer Wechselplatte sind verfügbar. Übrigens kann man in alle DataJets sehr leicht ein zweites SCSI-Laufwerk einbauen; das separat zu erstehende "Service-Handbuch" gibt dazu Hinweise.

Die Kompatibilität der Software zu AHDI V3.0 sollte Vortex verbessern, halbe Sachen lohnen sich nicht. Warum die neuen Partitionentypen einbauen, wenn man gleichzeitig die AHDI3-übliche Methode zur Kommunikation mit Cache-Programmen vernachlässigt (mit bombigen Folgen, siehe oben)?

Vortex möchte der DataJet-Reihe ganz offensichtlich das Image der Zuverlässigkeit verleihen; dafür spricht nicht nur das stabile Gehäuse und die saubere Bauart, sondern auch das GS-Zeichen des TÜV, die Funkentstörungsprüfung und der bereits erwähnte offizielle Geräuschtest. Vielleicht wäre es keine schlechte Idee, dem angestrebten Image mit einer etwas längeren Garantiezeit (derzeit nur 6 Monate) noch mehr Glaubwürdigkeit zu verleihen. Eine aufregende Platte ist der DataJet nicht, dafür aber ein ruhiges, zuverlässiges, doch keineswegs gemächliches Arbeitspferd. 1998 DM kostet der DataJet X60 - wahrlich kein niedriger Preis für eine 58-MB-Platte. Für sein Geld erhält man ein inzwischen ausgereiftes Produkt, das im Betrieb relativ wenig Ärger macht, sicher arbeitet und dank des Microboards einige Extras bietet: Motorabschaltung und Lüfterregelung, flexible Einschaltregelung und Hardware-Schreibschutz. Das mag in bestimmten Einsatzbereichen den Ausschlag für die Kaufentscheidung geben.

CB

Literatur:

[1] Bohs/Brod/Stepper: "Ganz schön dicht", ST-Computer 1/90

SuperCharger mit AT-Power?

Die Odyssee geht weiter

Im Oktober 1989 stellten wir Ihnen den SuperCharger vor. Dieses Gerät emuliert einen MS-DOS-Rechner. Inzwischen ist es weiter verbessert worden; viele Erleichterungen sollen dem Anwender das Leben noch leichter machen, als es ohnehin schon war. So sind in der neuen Version 1.4 der Software einige Fähigkeiten hinzugekommen, von denen andere Emulatoren nur träumen können.



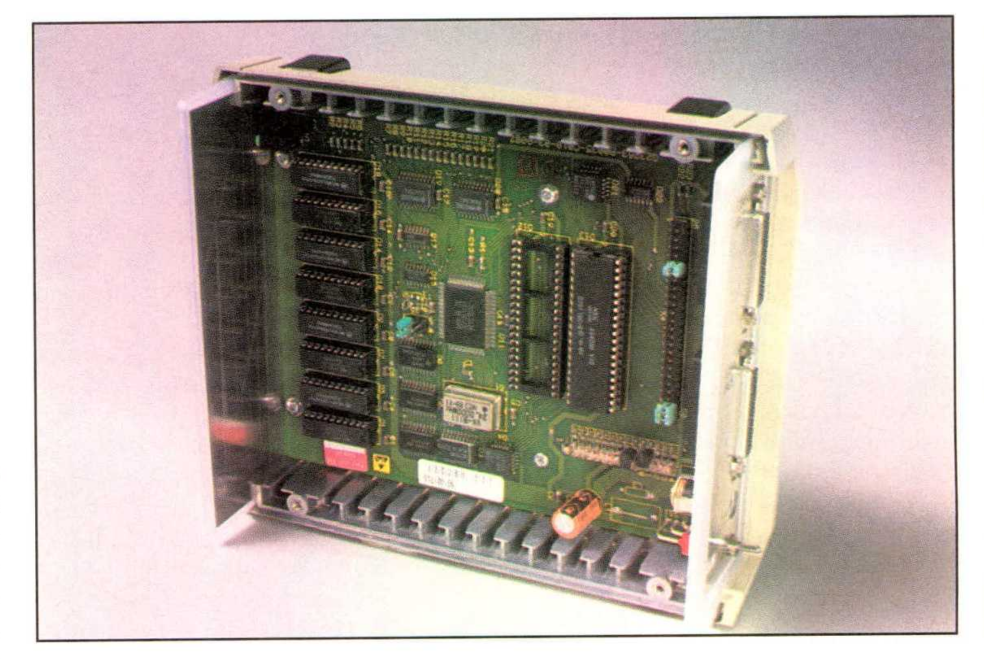
Der SuperCharger ist bislang der einzige MS-DOS-Emulator, der direkt angesteckt werden kann. Es muß also kein Eingriff in den Rechner durchgeführt werden, sondern der Mini-PC läßt sich direkt in den DMA-Bus des ST einstecken. Dadurch werden Lötarbeiten unnötig - ein großer Vorteil gegenüber dem Rest der am Markt erhältlichen Emulatoren.

Zuerst jedoch eine kleine Übersicht über die Fähigkeiten des Geräts für alle die, die den Artikel in der Oktober-Ausgabe des Jahrs 1989 nicht lesen konnten bzw. können.

SuperCharger besitzt einen Norton-Faktor von 4,0. Der Performance-Test von Richard B. Johnson bescheinigt ihm eine Performance (ich würde das Wort gerne übersetzen, nur wie?) von 297 Prozent. Alle wichtigen getesteten Programme funktionierten einwandfrei.

Genialer Hotkey

Damit sind die Gleichheiten zum Oktober '89 allerdings auch schon abgehandelt, denn der Rest des Geräts ist ziemlich neu, ja sogar brandheiß. Vielleicht haben Sie bei anderen Emulatoren schon festgestellt, daß sie einen "Hotkey" besitzen. Während man beim SuperCharger mit der Tastenkombination Alt-Control-Backspace in den ST-Modus zurückschalten kann und diese Funktion als Hotkey bezeichnet wird, dient die Hotkey-Funktion bei anderen Emulatoren lediglich zum Invertieren der Bildschirmfarbe. Hier liegt die erste Neuerung des SuperChargers, denn ab sofort ist auch diese (unwichtige) Funktion implementiert: Mit Alternate-S-A schaltet man auf die inverse Darstellung um (Switch to Alternate Color Key). Dieser kleine Unterschied zwischen den Hotkey-Funktionen ist übrigens im Handbuch recht amüsant beschrieben.



Neben dieser Funktion, die für einen Emulator alles andere als wichtig ist, hat man natürlich auch "vernünftige" Erweiterungen eingebaut. Durch Alternate-Control-Backspace gelangt man, wie bereits gesagt, in den ST-Modus zurück. Während dieser Zeit wird der SuperCharger gestoppt und arbeitet nicht weiter. Wird nun das Programm *ABIO.TOS* gestartet, befindet man sich wieder im MS-DOS-Modus. Das Programm befindet sich dabei genau an der Stelle, an der es gestoppt wurde. Das ist immer noch keine Neuerung, aber auch nicht bei jedem Emulator zu finden. Die wichtigste Neuerung ist, daß das *ABIO.TOS*, also quasi das Betriebssystem des SuperChargers im ST selbst, nun auch als Accessory installiert werden kann. Schön, werden Sie sagen, das ist immer noch nicht sonderlich neu. Was allerdings passiert, wenn man Shift-Control-Backspace drückt, sprengt die Vorstellungen aller bisher dagewesenen Emulatoren: echtes Multitasking mit dem ST!

Multitasking

Drücken Sie die oben beschriebene Tastenkombination, springt SuperCharger zurück in den ST-Modus. Hier kann nun, wie gewohnt, weitergearbeitet werden. Die Neuerung (jetzt kommt sie endlich) dabei ist, daß der SuperCharger im MS-DOS-Modus im Hintergrund weiterarbeitet! Das heißt im Klartext: Zwei Rechner arbeiten gleichzeitig. Wollte man früher eine Datenbank erstellen, die man beispielsweise mit dBASE und Adimens bearbeiten wollte, verlief es folgendermaßen: MS-DOS laden, dBASE laden, Datenbank erstellen, zwischendurch einen Kaffee trinken, nach Beendigung in den ST-Modus, die entstandene Datenbank in Adimens verarbeiten, zum Schluß einen erläuternden Text zur Datenbank schreiben. Ein anderes Beispiel: Sie müssen 10 Texte schreiben und alle ausdrucken. Also verfahren Sie wie folgt: Schreiben, Drucken und Warten; Schreiben, Drucken und Warten.....

Wie funktioniert es denn jetzt genau? Ganz einfach: MS-DOS aktivieren, dBASE laden, Datenbank erstellen lassen. Nun fällt allerdings die Kaffeepause weg, denn während (!) die Datenbank erstellt wird, schaltet der SuperCharger-Benutzer um in den ST-Modus und schreibt bereits jetzt einen erläuternden Text oder spielt eine Runde Tetris, natürlich kann man auch schnell in einer Mailbox anrufen oder seine Festplatte reorganisieren; die Möglichkeiten dieser Anwendung finden keine Grenzen. Das andere Beispiel ist ebenso leicht auf diese Anwendung zu übertragen: Sie schreiben im ST-Modus einen Text. Sobald Sie fertig sind, wechseln Sie in den MS-DOS-Modus und lassen den Text über den eingebauten Drucker-Spooler ausdrucken. Während das geschieht, können Sie bereits auf dem ST den nächsten Text bearbeiten. Durch diese Möglichkeit erlangt der SuperCharger einen erheblichen Vorsprung zu anderen Emulatoren - bislang beherrscht kein anderer das Multitasking. Das wird auch so schnell nicht der Fall sein, da der SuperCharger der einzige ist, der ein eigenes RAM dieser Größe (ab 512 kB) besitzt.

MS-DOS im TOS

Stellen Sie sich folgende Situation vor: Sie befinden sich im ST-Modus und klicken mit der Maus auf die Datei "PCTOOLS.EXE". Was würde ein "normaler" ST dazu sagen? Ganz einfach: "*Diese Datei kann nur angezeigt oder gedruckt werden*". Nun existiert im SuperCharger die Möglichkeit, ein *LAUNCH.TTP* zu installieren, das auf Dateien mit den Endungen "COM", "EXE" und "BAT" anspricht. Was soll überhaupt passieren: *PCTOOLS.EXE* wird angeklickt. Dann soll der Emulator gestartet werden, der wiederum von sich aus PCTOOLS starten soll. Nachdem PCTOOLS beendet ist, soll der Emulator automatisch beendet werden und zurück in den TOS-Modus gehen. Sie denken, das ist nicht möglich? Und doch, mit dem SuperCharger kein Problem. Sofern das Programm *LAUNCH.TTP* installiert ist, schaltet der ST immer dann in den MS-DOS-Modus um, wenn eine EXE-, COM- oder BAT-Datei angeklickt wird. Nach deren Abarbeitung gibt der Emulator die Arbeit automatisch an den ST zurück. Dazu kann man eigentlich nur ein Wort sagen: genial! Wer einmal mit dieser Funktion gearbeitet hat, möchte den Komfort auf keinen Fall mehr missen.

Noch mehr

"SuperCharger mit AT-Power?" heißt die Überschrift dieses Artikels. Natürlich sitzt im Gehäuse des Geräts noch immer ein NEC V30-Prozessor, im Gegensatz zu einem AT, der ja normalerweise mit einem 80286-Prozessor ausgerüstet ist. Wieso also "AT-Power"? Nun, ein 80286 besitzt gewöhnlicherweise einen Satz von Möglichkeiten, der im BIOS (Basic Input-/Output-System) festgehalten ist. Die Werbung würde sagen: "Herkömmliche Waschmittel haben..." - ich sage: herkömmliche Emulatoren haben das AT-BIOS nicht eingebaut. Das hat der SuperCharger ebenfalls nicht, ein großer Teil des BIOS ist jedoch mit dem des AT-BIOS kompatibel, von daher darf man sicherlich behaupten, daß der SuperCharger ein Quasi-AT-Emulator ist. Wo viele Firmen auf den Prozessor setzen, das BIOS aber vernachlässigen, zeigt Beta Systems korrekte Ansätze: Das BIOS

stimmt einfach. So läuft beispielsweise auch *Windows 286* problemlos mit dem SuperCharger, was bestimmt nicht jeder MS-DOS-Emulator von sich behaupten kann. Sicher kann man sich denken, daß somit auch andere Programme, die ursprünglich für einen "echten" 80286-Rechner konzipiert sind, ihre Funktion nicht verweigern werden.

Viele weitere Neuigkeiten warten auf denjenigen, der den Charger gebrauchen will: Endlich kann auch die serielle Schnittstelle benutzt werden. Die Atari-Maus liegt weiterhin auf COM1 und belegt damit dauerhaft eine Schnittstelle, was jedoch nicht weiter schlimm ist, da nun über COM2 auch die serielle Schnittstelle des ST angesprochen werden kann. Die Atari-Maus ist als Mouse Systems-kompatible Maus ansprechbar. Bis zu einer Übertragungsgeschwindigkeit von 9600 Baud konnten keine Probleme mit COM2 festgestellt werden. Selbst Telix, das Shareware-DFÜ-Programm für den PC schlechthin, funktionierte einwandfrei und gab keinerlei Grund zum Ärgernis. Telix ist sehr hardware-nah programmiert und greift direkt auf einige Bauteile zu, die im SuperCharger eigentlich nicht vorhanden sind - trotzdem funktioniert es einwandfrei. Wo wir gerade bei Schnittstellen sind: Der ATARI-Laser kann übrigens auch angeschlossen werden.

Verfügte der SuperCharger in der Oktober '89-Version nur über MGA- und CGA-Emulation, kann jetzt auch die Hercules-Auflösung benutzt werden. Dabei ist man einen ähnlichen Weg wie bei PC-Speed gegangen: Da die Hercules-Auflösung höher als die des ST ist, kann man den Bildschirm entweder nach links, in die Mitte oder nach rechts schieben. Dazu genügt ein Druck auf die "/"-Taste auf dem Ziffernblock. Die anderen Auflösungen haben keine Veränderung erfahren und funktionieren natürlich weiterhin problemlos.

Toolbox

Unter dem Namen "Toolbox" wird seit Version 1.30 auch eine Möglichkeit für Programmierer mitgeliefert, den SuperCharger für eigene Anwendungen zu nutzen. Das ist auch sinnvoll, denn wieso sollte man das SuperCharger-RAM nicht nutzen, wenn man es schon auf dem Schreibtisch stehen hat? Eine beispielhafte Anwendung wird auch bereits mitgeliefert, eine RAM-Disk, die vom ST aus zu nutzen ist, soll dem Anwender das Leben erleichtern. Mit der RAM-Disk läßt sich vom ST aus auf das RAM des SuperChargers zugreifen. So wird der Speicher des ST um eine beträchtliche Größe erweitert. Der Vorteil der RAM-Disk: Sie ist schnell, zuverlässig, groß und außerdem reset-fest! Das heißt: Solange dem SuperCharger der Strom nicht genommen wird, bleiben die Daten in seinem Speicher erhalten. Auch wenn der ST resettet wird, spielt es keine Rolle: Alle Daten sind nach wie vor enthalten, lediglich den Treiber für die RAM-Disk müssen Sie neu laden.

Daraus ergeben sich natürlich einige Anwendungen, die ohne den SuperCharger nicht möglich wären. Das Multitasking sprach ich ja bereits weiter oben an, so allerdings erhält das Wort ein weiteres Mal eine neue Bedeutung. Läßt man im Hintergrund nicht den Emulator laufen, sondern bedient sich "nur" des V30-Prozessors, lassen sich nette Effekte erzielen. Apfelmännchen, die wirbelwindartig über den Bildschirm huschen, Primzahlenberechnung in Windeseile, Paßwortsicherung im SuperCharger-RAM oder eben "nur" die oben erwähnte RAM-Disk sind lediglich einige Anwendungsgebiete, die sich durch die Toolbox eröffnen. Sicher dürfte es auch ein leichtes sein, sich aus dem SuperCharger ein intelligentes Cache-Programm für seine Festplatte zu schreiben - wer hätte schließlich nicht gerne 0,5 oder 1 MB Cache mit eigenem V30-Prozessor?

Beispiele für Anwendungen, die im, um oder am SuperCharger funktionieren, sind auf Diskette und im Handbuch enthalten, so daß dem erfahrenen Programmierer keine Probleme bei der Programmierung widerfahren dürften. Allerdings sollte man zumindest in C oder Assembler programmieren können, um die Funktionen adäquat ausnutzen zu können.

Schlußbetrachtung

Man kann nicht oft genug darauf hinweisen: Der SuperCharger hebt sich durch die exzellente Implementation des "Hotkeys" von allen anderen Mitbewerbern mehr als positiv ab. Ebenso kann sich das BIOS des Testgeräts sehen lassen, da es einen AT-kompatiblen MS-DOS-Rechner nachahmt, und der SuperCharger dadurch die Bezeichnung "AT-Emulator" besser verdient als "PC-Emulator". Außerdem wird, und das ist längst nicht der Regelfall, MS-DOS 4.01 mitgeliefert - dieses Betriebssystem allein kostet bereits einige hundert Mark. Das Gerät ist einfach ansteckbar und muß nicht eingelötet werden - ein weiterer Vorteil gegenüber anderen Emulatoren. Der Preis von DM 875,- mutet auf den ersten Blick recht hoch an. Bedenkt man aber, daß MS-DOS 4.01 mitgeliefert wird, ist der Preis doch sehr günstig. Zum Schluß sollten Sie einen Vorteil nicht außer Acht lassen: Der SuperCharger läuft tadellos auf dem TT! Kein anderer MS-DOS-Emulator kann das bis jetzt von sich behaupten.

MP

DEMO DISKS

Demo-Disketten kosten lediglich DM 10,- **pro Diskette** und können über die Redaktion bezogen werden. So müssen Sie zum Vergleich verschiedener Programme nicht an verschiedene Hersteller schreiben, sondern können sich in aller Ruhe das Demonstrationsprogramm ansehen, bevor Sie das Original kaufen.

Bitte beachten Sie, daß die angebotenen Disketten nur Demonstrationsdisketten der Originalversionen sind und somit im Gegensatz zu den Originalen in Funktion eingeschränkt sind!

z.Zt. erhältlich:

D1: S.&P.-Charts
Chart-Analyseprogramm
(*S.P.S. Software*)

D2: SPC-Modula-2
Modula-2-Entwicklungssystem
(*Advanced Applications Viczena*)

D3: ST-Fibu
Finanzbuchhaltungsprogramm
(*GMa-Soft*)

D4: ST-Fibu-Fakt
Fakturierungsprogramm für ST-Fibu
(*GMa-Soft*)

D5: ST-Fibu-Text
Textverarbeitungsprogramm für ST-Fibu mit Serienbrieffunktion
(*GMa-Soft*)

D6: SciGraph
Programm zur Erstellung von Präsentationsgrafiken (*SciLab GmbH*)

D7: ST-Statistik
Uni- und multivariates Statistikprogramm, Grafikeinbindung (*SciLab GmbH*)

D8: fibuSTAT
Finanzbuchhaltungs-/Statistikprogramm
(*novoPLAN Software GmbH*)

D9: Btx/Vtx-Manager
Programm zum Anschluß an Bildschirmtext
(*Drews Btx + EDV GmbH*)

D10: Edison
Editor für fast alle Gelegenheiten
(*Kniss Soft*)

D11 & D12: CADjA
CAD-Programm für hohe Ansprüche
(*Computer Technik Kieckbusch*).
Demo besteht aus zwei Disketten zu je DM 10,-!

D13: JAMES 2.0
Programm für Börsenspekulanten
(*IFA-Köln*)

D14: Soundmerlin
Sample-Editor-Programm mit vielen Modulen
(*TommySoftware*)

D15: Soundmachine II
Programm zur Erstellung und Wiedergabe von Sounds
(*TommySoftware*)

D16: ReProK
Büroorganisationsprogramm
(*Stage Microsystems*)

D17: Sherlook
Schrifterkennungs- und -verarbeitungsprogramm (*H.Richter*)

D18: ST Matlab
Programmiersystem mit Schnittstelle zu Modula-2 (*Advanced Aplications Viczena*)

D19: Calamus
Desktop-Publishing-Programm
(*DMC*)

D20: GD-Fibu
Finanzbuchhaltungsprogramm
(*GDAT*)

D21: Omikron.Draw!
Zeichen- und Malprogramm
(*Omikron.Software*)

D22: Omikron.Libraries
Verschiedene Libraries für Omikron.BASIC (*Omikron.Software*)

D23: Omikron.Compiler
Demo-Version des Omikron.BASIC-Compilers (*Omikron.Software*)

D24: Mortimer
Multi-Programm für alle Gelegenheiten
(*Omikron.Software*)

D25: Script
Textverarbeitungsprogramm
(*Application Systems /// Heidelberg*)

D26: SuperScore
Sequencer- und Notendruckprogramm
(*BELA Computer GmbH*)

D27: SPS-Emulator
Programm zur Simulation von SPS-Steuerungen
(*Karstein Datentechnik*)

D28: STAD 1.3+
Zeichenprogramm mit 3D-Teil
(*Application Systems /// Heidelberg*)

D29: MegaFakt
Fakturierungsprogramm
(*MegaTeam*)

D30 & D31: MegaPaint II
Zeichenprogramm mit Vektorteil
(*TommySoftware*)

D32: Tempus Word
Textverarbeitung
(*CCD*)

D33: Creator
Zeichenprogramm mit Animationsteil
(*Application Systems /// Heidelberg*)

D34: Outline Art
Utility für Calamus
(*DMC*)

D35: compugraphic Schriften
für Calamus
(*DMC*)

D36: BTX-Börsen-Manager
Börsenprogramm
(*Thomas Bopp Softwarevertrieb*)

D37: Cashflow
Kassenbuch
(*C.A.$.H.*)

D38: TiM II
Finanzbuchhaltungsprogramm
(*C.A.$.H.*)

D39: Merkator
(Finanzbuchhaltungsprogramm)
(*MDC*)

D40: Technobox Drafter
(Zeichenprogramm spez. f. Konstruktionen)
(*Technobox*)

Es gelten die gleichen Vertriebsbedingungen wie für PD-Disketten (s. PD-Seiten am Ende dieser Ausgabe). Demo-Disketten können auch zusammen mit PD- und Sonder-Disketten bestellt werden.

Bitte vergessen Sie nicht die betreffende Bestellnummer (z.B. D1) anzugeben.


VERSION: 3.7= Dreidimensionales
File-Name = DREIZYMO.C3D
EIZO FLEXSCAN

MGE maxon graphic expansion

Mit ihrem schnellen Intel 82786-Grafikprozessor und ihren vielfältigen Auflösungen bietet die MAXON Graphic Expansion (MGE) die ideale Profi-Grafikerweiterung für alle Mega ST-Besitzer mit geeignetem Monitor (z.B. MultiSync oder Großbildschirm).

Sie besticht durch ihren Leistungsumfang und ihre Flexibilität. Durch den GDOS-VDI-Treiber sind GEM-Programme lauffähig. Zusätzlich wird eine S/W-Großbildschirm-emulation mitgeliefert. Egal ob Sie Desktop Publishing (z.B. Calamus), CAD oder anspruchsvolle Farbgrafik wie z.B. Desktop Video einsetzen wollen, die MGE verwandelt Ihren Mega ST in eine Grafik-Workstation!

Auflösungen:

1. 13,75 MHz, 800 x 600 Bildpunkte, maximal 256 Farben interlace, 50 Hz Halbbildfrequenz (für Video- und Genlock-anwendungen);
2. 27,5 MHz, 640 x 480 Bildpunkte, maximal 256 Farben aus 262 144 non-interlace, 66 Hz Bildfrequenz;
3. 27,5 MHz, 800 x 600 Bildpunkte, maximal 256 Farben aus 262 144 interlace, 80 Hz Halbbildfrequenz;
4. 55 MHz, 896 x 684 Bildpunkte, maximal 16 Farben aus 262 144 non-interlace, 67 Hz Bildfrequenz;
5. 55 MHz, 1280 x 1024 Bildpunkte, maximal 16 Farben aus 262 144 interlace, 65 Hz Halbbildfrequenz;
6. 110 MHz, 1280 x 960 Bildpunkte, monochrom non-interlace, 66 Hz Bildfrequenz.

Mitgelieferte Software:

- Hardware-Treiber
- GDOS-VDI-Treiber
- VDI-Druckertreiber
- Monitorbibliotheken
- Kontrollfeld zur Einstellung der Auflösung und Farben
- Bindings für Turbo-C, Assembler und GFA-BASIC

Preise:

MGE II
Bestell-Nr.: 890810 ....DM 2398,- *

Erweiterung auf 16,7 Mio. Farben
Bestell-Nr.: 890816 .... NEU! ....DM 79,- *

Arithmetik-Coprozessor 68881
Bestell-Nr.: 890815 ....DM 348,- *

Komplettsysteme:

MGE II + Eizo 9070S (16"-Farbmonitor)
Bestell-Nr.: 890812 ....DM 5498,- *

MGE II + Eizo 6500 (21"-Graustufengroßbildschirm)
Bestell-Nr.: 890813 ....DM 6298,- *

MGE II + Eizo 9400 (20"-Farbgroßbildschirm)
Bestell-Nr.: 890814 ....DM 8998,- *

* unverbindlich empfohlene Verkaufspreise

Lassen Sie sich die MGE bei folgenden Händlern vorführen:

MCC
Holzkoppelweg 19a
2300 Kiel 1

PS-DATA
Hard- und Software GmbH
Faulenstr. 48-52
2800 Bremen

COM-DATA GmbH
Schiffgraben 19
3000 Hannover 1

3K Computerbild GmbH
Sassenfeld 71
4054 Nettetal 1

CSA Computersysteme
Hüttenstr. 56
4650 Gelsenkirchen

Data Pach GmbH & Co KG
Neuhausstr. 4
5100 Aachen

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90

Pauly Büromaschinen
Salzgasse 6
6250 Limburg/Lahn

Schreiber Computer
Rotebühlplatz 10
7000 Stuttgart 1

Walliser & Co KG
Marktstr. 48
7000 Stuttgart 50
Bad Cannstatt

Comp & Phone
Alleenstr. 66
7312 Kirchheim-Teck

GCS Röseler
Jahnstr. 1
7343 Kuchen
Tel.: 07331/81925
(nach Vereinbarung)

Besch & Partner
Stuttgarter Str. 53
7432 Bad Urach

JOBI'S
Unternehmenssupport
DV & Marketing
Fliegenstr. 12
8000 München 2

HIB Computer
DTP-Center
Äußere Bayreuther 57a-59
8500 Nürnberg 10

CCN Computer Center
Nürnberg GmbH
Giebitzenhofstr. 86
8500 Nürnberg 70

V. Willgeroth
Bürobedarfshaus
GmbH & Co KG
Hanauer Str. 12
8750 Aschaffenburg

Händleranfragen erwünscht!

MAXON Computer
Schwalbacherstr. 52
6236 Eschborn
Tel.: 0 61 96 / 4818 11
FAX: 06196/41885



PANORAMA-BLICK Overscan

Die Auflösungs-Erweiterung

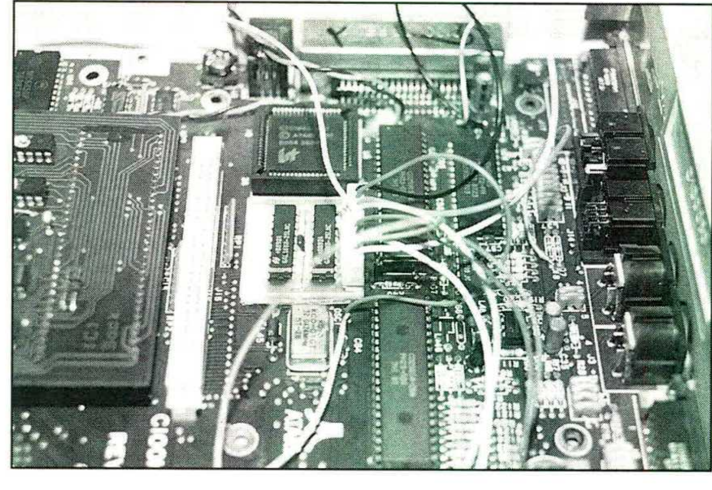


Bild 1: Die Overscan-Platine in einem MEGA ST

Mit einer originellen Idee und einer einfachen Schaltung, garniert mit einem Stück Software, läßt sich die Bildschirmauflösung des ST massiv erhöhen. Das Produkt nennt sich Autoswitch-Overscan und ist für DM 97,- erhältlich.

Overscan hat eine längere Entwicklungsgeschichte und tauchte erstmalig unter dem Namen Hyperscreen in einer ST-Zeitschrift auf. Damals waren als Hardware nur ein paar Kabel und ein Umschalter notwendig. Die Software war frei und wurde mehrfach überarbeitet.

Die Entwickler haben sich nun hingesetzt und aus der Idee ein professionelles Produkt gemacht. Herausgekommen ist der Autoswitch-Overscan, der die Auflösung des ST massiv erhöht und sich durch eine kleine Logik automatisch auf nicht auflösungsunabhängig programmierte Software einstellt.

Die mit Overscan erzielbaren Auflösungen sind phänomenal und sollen selbst die ATARI-Entwickler überrascht haben. Ein normaler ST mit der Overscan-Schaltung kann monochrom bis über 700 * 480, in der mittleren Auflösung bis über 800 * 280 und in der niedrigen Einstellung bis zu 400 * 232 Pixel darstellen.

Wie hoch die Auflösung getrieben werden kann, hängt vom verwendeten Monitormodell und teilweise auch von geräteeigenen Eigenschaften ab. Die Entwickler geben beispielsweise für einen SM124 von ATARI eine monochrome Auflösung von 672 * 480 an, ein NEC Multisynch GS bringt sogar 752 * 480. Mit ATARIs SC1224 kommt man auf 384 * 280 bzw. 752 * 280 - ein Multisynch oder ein AMIGA-Monitor erreicht dagegen 400 * 280 bzw. 816 * 280 in Farbe. Ein Test mit einem billigen portablen Fernseher erbrachte noch 352 * 272 bzw. 736 * 272 Pixel.

Möglich wird das Ganze, indem - grob gesagt - der Shifter einen höheren Takt erhält. Dieser wird einfach an einer anderen Stelle auf der ST-Platine abgegriffen und von der Hardware mit dem Shifter verbunden. Aufgrund der eingebauten Logik in zwei GAL- Bausteinen kann die Verbindung und damit die Auflösung unter Software-Kontrolle geschaltet werden. Der Software-Treiber kümmert sich im Kern um die Verwaltung des neuen Bildschirmspeichers, erlaubt die Umschaltung der Auflösung und bindet das Ganze in GEM ein.

Overscan kann mit allen ST-Modellen mit Ausnahme des 1040STE verwendet werden. Bei diesem Modell fiel ein benötigtes Signal der Integration zum Opfer und ist nicht mehr zugänglich, sondern direkt auf einem Chip verschaltet.

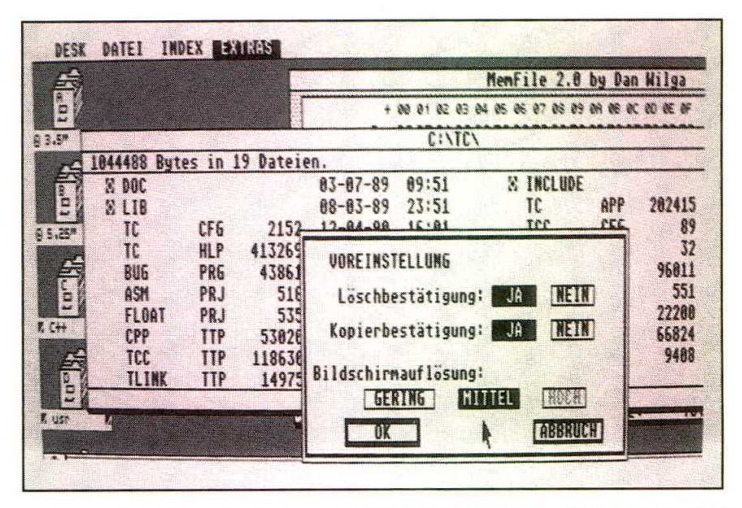
Einsteigen

Am Anfang steht der Einbau des kleinen Platinchens, auf dem sich die beiden GAL-Bausteine befinden. Wie in Bild 1 zu sehen, erhebt sich ein kleiner Kabelbaum aus der Schaltung, denn immerhin müssen zehn Verbindungen an verschiedene Stellen auf der ST-Platine gezogen werden.

Neben dem Auffinden dieser Punkte und der Lötarbeit sind noch drei Leiterbahnen auf der ST-Platine zu durchtrennen. Aufgrund eines Layout-Fehlers muß bei einigen MEGA-Modellen ein weiterer Kontakt durchtrennt werden. Sind die Verbindungen gezogen, kann das Overscan-Platinchen im Rechner verstaut werden, was durch einen einfachen Haftstreifen unterstützt wird. Im Bild hat Overscan neben dem 68000 Platz gefunden und gerät auch nicht mit einem PC-Speed ins Gehege. Sollten andere Zusatzplatinen vorhanden sein, ist die Plazierung der Overscan-Schaltung ebenfalls problemlos.

Der Einbau ist nicht unbedingt für Anfänger geeignet, da die entsprechenden Lötpunkte auf der Platine natürlich exakt verbunden werden müssen und die Leiterbahnen auf einer modernen Platine wie der des ST sehr fein sind. Traut man sich den Einbau nicht zu, sollte man ihn besser einem Händler oder einem versierten Bekannten überlassen.

Das 48seitige Handbuch beschreibt den Einbau detailliert in 18 Schritten und gibt einige Tips, falls nicht sofort das gewünschte Ergebnis eintritt. Mehrere Grafiken beschreiben die Lage der durchzutrennenden Leiterbahnen und der Lötpunkte für die verschiedensten ST-Modelle und -Platinen. Das Handbuch sollte keine Fragen zum Einbau offen lassen.



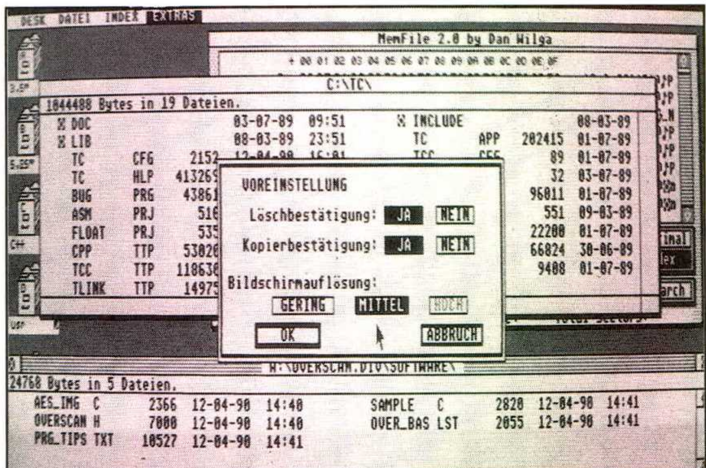


Bild 2: Beeindruckend: originale und erweiterte Auflösung im mittleren Modus

Hochfahren

Der Software-Treiber OVERSCAN.PRG muß von der mitgelieferten Diskette in den AUTO-Ordner kopiert werden. Damit wird bei jedem Booten automatisch die Overscan-Darstellung eingeschaltet. Beim ersten Start muß die Software auf den verwendeten Monitor konfiguriert werden, wozu auf Festhalten einer Shift-Taste beim Booten ein Menü erscheint.

Ein Rechteck zeigt dann den momentan von Overscan genutzten Bildschirmbereich an. Er kann mit den Pfeiltasten so vergrößert oder verkleinert werden, daß er vollständig auf dem Monitor angezeigt wird. Mit dem Zehnerblock läßt sich auch die Lage des angezeigten Monitor-Bilds verändern, so daß eine Anpassung an die Fähigkeiten des Monitors einfach und interaktiv möglich ist.

Weiterhin existieren verschiedene Optionen bei der Konfiguration. So können bei einem Farbmonitor die Auflösung, die Darstellungsfrequenz und die Farbe des Bildschirmrands gewechselt werden

Da die ST-interne Hardcopy-Routine ihre Macken hat und mit einem großen Bildschirmspeicher nicht umgehen kann, läßt sich Overscan so einstellen, daß beim Auslösen einer Hardcopy der gesamte Overscan-Bildschirminhalt in eine Datei im GEM-Image-Format geschrieben wird. Im Schwarzweißbetrieb läßt sich die Auflösungsumschaltung durch eine Zusatzspeicheroption beschleunigen.

Hat man alle Einstellungen nach Wunsch getätigt, werden sie in OVERSCAN.PRG abgespeichert und stehen beim nächsten Booten sofort bereit. Will man - aus welchen Gründen auch immer - nur kurzzeitig eine andere Bildschirmauflösung benutzen, kann das Konfigurations-Menü auch ohne Abspeichern verlassen werden.

Abfahren

Beim nächsten Booten fährt der Treiber dann die Bildschirmauflösung mit einem optischen Effekt auf die konfigurierten Werte hoch. Es erscheint das Desktop, und der seinen festen Bildschirmrahmen gewohnte Benutzer wird zunächst einmal den Atem anhalten. Bild 2 zeigt das gleiche Desktop mit und ohne Overscan- Auflösung auf einem Farbmonitor. Neben dem offensichtlichen Auflösungsgewinn (in den Farbauflösungen läßt sich die Pixel-Anzahl glattweg verdoppeln) sieht man auch, daß hier mit einem normalen ST auch die Mehrspaltendarstellung der Directory- Fenster möglich ist. Ohne Übertreibung - man fühlt sich fast wie auf einem Großbildschirm.

Abbremsen

Leider hat sich schon mit dem Aufkommen von Großbildschirmen oder dem Software-Emulator BigScreen gezeigt, daß viele ST- Programme nicht auflösungsunabhängig programmiert sind. Dies reicht von einfacheren Problemen wie festen Maximalgrößen für Fenster bis zu Programmen, die direkt in den - vermeintlich normalen - Bildschirmspeicher schreiben. Letztere liefern bei veränderter Bildschirmgröße wilde Bit-Muster und sind in einer solchen Konfiguration nicht zu gebrauchen. Ursache dafür ist sicher das Fehlen fester ATARI-Richtlinien für die Programmierung, wie sie beispielsweise Apple für seinen Macintosh durchgesetzt hat.

Dieses Problem wird sich nicht nur bei Systemen mit Großbildschirmen bemerkbar machen, sondern auch auf den neuen ATARI-Maschinen TT und EST, die bekanntlich erheblich mehr Auflösungsmodi bieten. Aber auch auf einem mit Overscan ausgerüsteten normalen ST machen einige Programme Schwierigkeiten. Das Problem wird gelöst, indem man Overscan einfach mitteilt, welche Programme nicht in der erweiterten Auflösung laufen. Ruft man solche Software vom Desktop, aus einer Shell oder anderen Programmen auf, schaltet Overscan automatisch, ohne daß irgendwelche Eingaben nötig wären, auf die alte Originalauflösung. Wird ein auflösungsabhängiges Programm beendet, tritt wieder der erweiterte Overscan-Modus in Kraft.

Festgelegt wird der gewünschte Auflösungmodus in einer ASCII-Datei OVERSCAN.INF, die beim Booten eingelesen wird. Dort könnte man beispielsweise mit einem Eintrag "GULAM.PRG -" auswählen, daß die Shell Guläm nicht in hoher Auflösung gefahren wird. Übrigens bleibt hier die eingestellte Auflösung beim Aufruf weiterer Programme aus der Shell erhalten - es sei denn, man hätte beispielsweise mit einem Eintrag "WORDPLUS.PRG +" festgelegt, daß WordPlus immer in der höheren Auflösung gestartet wird.

Bei der Erstellung der Einträge kann man übrigens auch Listen mit Programmnamen schreiben, für die eine Einstellung gemeinsam gelten soll, oder Wildcards verwenden. Die .INF-Datei wird beim Einlesen gepackt, wofür ein Datenraum von maximal vier KB bereitsteht. Gerade durch die Wildcard dürfte dies ausreichen.

Man wird also während der Arbeit mit seinen Programmen feststellen, ob ein

Programm auflösungsunabhängig ist. Trifft dies nicht zu, trägt man einfach den Namen mit einem Editor in die Info-Datei ein, und nach dem nächsten Booten stellt Overscan automatisch den gewünschten Modus ein. Will man sofort mit einem problematischen Programm arbeiten, kann man durch Drücken von Alternate +Control beim Doppelklick auf dem Desktop manuell die original ST-Auflösung anwählen. Diese Tastenkombination läßt sich übrigens durch einen im Handbuch beschrieben Patch mit einem Disk-Monitor ändern.

Hochgetrieben

Der Software-Treiber muß bei der Umschaltung der Auflösung eine ganze Reihe von Betriebssystemvariablen verändern. Gleichzeitig hängt er sich in fast alle System-Trap-Vektoren ein, um das Verhalten einiger Funktionen zu ändern. Der Treiber arbeitet mit allen TOS-Versionen, darunter auch dem "durchgepatchten" KAOS. Lediglich bei der Uraltversion des ersten ROM-TOS sind Patches nötig.

Für den Programmierer implementiert der Software-Treiber zusätzliche XBIOS-Funktionen, mit denen beispielsweise die Versionsnummer abgefragt oder die Auflösung von einem Programm aus umgeschaltet werden kann. In einem Header-File für TURBO-C werden die Funktionen zugänglich gemacht.

Da Overscan auch implizit eine Testmöglichkeit für die Auflösungsunabhängigkeit eigener Programme ist, sollte unter Overscan selbsterstellte Software zu den verschiedensten Auflösungen kompatibel sein. Auf welche Feinheiten man in der Programmierung dabei zu achten hat, und mit welchen GEM-Funktionen man beispielsweise die X-Auflösung wirklich sicher abfragt, erläutert ein kurzer, aber sehr interessanter Text auf der Diskette. Zwei Beispielprogramme in GFA und C sowie C- Routinen zur auflösungsgerechten Verwendung von Image-Daten bei Icons runden die zusätzliche Software ab.

Übrigens läßt sich die erhöhte Auflösung beispielsweise mit pc- speed schon seit geraumer Zeit nutzen. Durch Auswahl der Bildschirminstallation "HYPERHERC" kann mit Overscan eine Hercules-Karte emuliert werden. Nach Angaben der Entwickler sollen auch der Mac-Emulator Spectre und der QL-Emulator an Overscan angepaßt werden.

Fazit

Das Erscheinen einer kostengünstigen Erweiterung, die eine erhöhte Bildschirmauflösung ermöglicht, könnte bei größerer Verbreitung auch Einfluß auf die Software-Qualität auf dem ST haben. Programme, die nur mit den üblichen Auflösungen arbeiten, werden unter Overscan sofort unangenehm auffallen und landen so in der Ausnahmeliste, die schließlich zu einer "schwarzen Liste" schlecht geschriebener Software werden kann.

Der Autoswitch-Overscan ist ein preiswertes Produkt, das den ST durch die erhöhte Auflösung deutlich aufwertet. Der Software- Treiber ermöglicht eine flexible Benutzung, und das Autoswitch-Konzept sichert auch die Verwendung auflösungsabhängig programmierter Software. Das Handbuch beschreibt Einbau und Betrieb eingänglich und sollte alle Fragen abdecken. Das kleine Overscan-Platinchen sollte und wird sicher seinen Weg in viele STs finden.

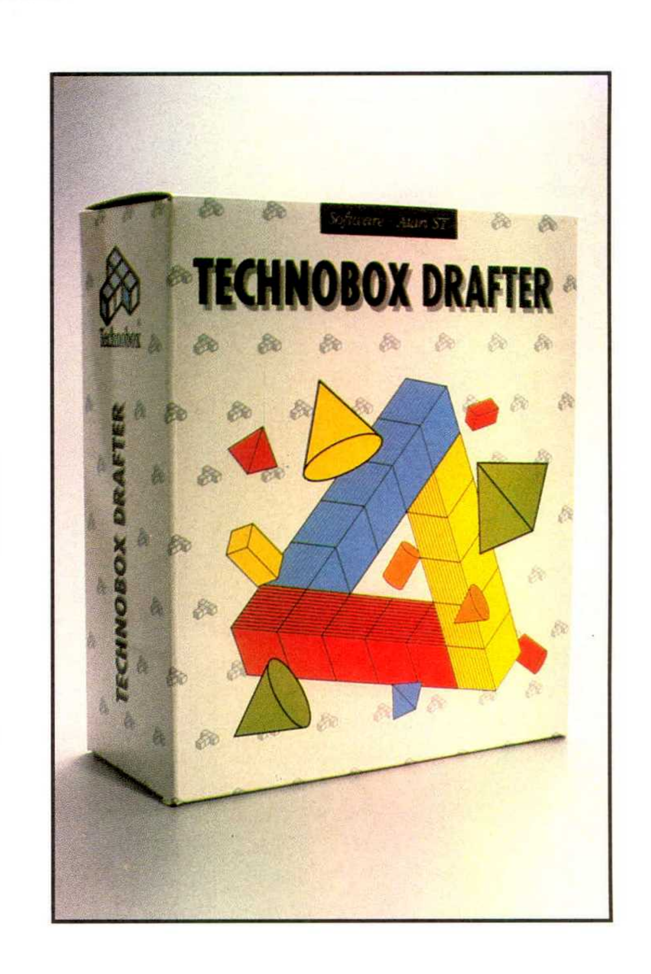
RT

Bezugsadresse:

Overscan GbR Isakovic, Hartmann, Jerchel
Säntisstr. 166
1000 Berlin 48

Neuauflage

Technobox Drafter



Wer auf der CeBIT das neue Produkt von Technobox gesehen hatte, fragte sich natürlich sofort: Was wird aus CAD/1, dem ehemaligen Campus? Das neue CAD/2 ist ja ausschließlich für professionelle Anwender gedacht, wofür vor allem der Preis verantwortlich ist. Kann der normale ST-User also nicht mehr bei Technobox kaufen?

Um die Antwort sind die Mannen aus Bochum nicht verlegen: Technobox Drafter, so heißt das Produkt, das man dem interessierten Heimanwender bereithält. Genaues Hinsehen verrät denn auch enge Verwandtschaft zum Vorgängerprodukt. Nichts Neues also? Ein wenig doch.

Für diejenigen, die die Namenspolitik des Hauses Technobox verwirrt, hier die Familiengeschichte: Ahn der CAD-Software aus Bochum ist das Programm 'Campus CAD', das es in den Versionen 1.0 bis 1.3 gab. Dann erfolgte aus rechtlichen Gründen eine Umbenennung in 'TechnoCAD'. Der Name wich schnell der Bezeichnung 'Technobox CAD/1', mit dem der Kunde bis heute (CAD/2 ist noch nicht auf dem Markt) ein CAD-System erhält, wie es professionellen Ansprüchen gerecht wird. Komplette Symbolbibliotheken und Software-Schnittstellen machen es zum umfangreichen Produkt. Dieses Programm testeten wir in der Juni-Ausgabe letzten Jahres [1].

Begrüßung

Wenden wir uns nun dem Drafter (deutsch: Zeichner) zu und packen den hübschen Karton aus. Eine schön aufgemachte und liebevoll gestaltete Anleitung in zwei Ringbüchern begrüßt den stolzen Besitzer. Alles in dezentem Grau - geschmackvoll. Der Text weist Anfänger wie Fortgschrittene in die hehren Weihen des Programms. Geduldig und präzise geht es dabei vor; man merkt, daß Erfahrung mit Anwendern in die Anleitung mit einflossen. Sicher wird auch derjenige schnell mit dem Programm umgehen können, der noch nie vor einem ATARI saß. Für meinen Geschmack könnte es etwas reicher illustriert sein, aber eine Wohltat sind die Register am Ende eines jeden Teiles.

Mit der Originaldiskette fällt eine der zweifelhaften Errungenschaften des Computerzeitalters aus dem Umschlag: ein Dongle. In den Joystickport eingesteckt, verhindert er die unerlaubte Benutzung des Programms. Daß man sich derartiger Mechanismen bedienen muß, wenn man mit Software Geld verdienen will, ist leider Realität. Doch wenn das Programm, wie im Test geschehen, mitten in der Arbeit den Dongle nicht mehr erkennt und das Speichern verweigert, ist das mehr als ärgerlich.

Die Oberfläche des Drafters ist ein wenig verändert gegenüber der des Vorgängers. Das Programm läuft jetzt komplett in einem GEM-Fenster, auch die Befehl-Icons befinden sich darin (siehe Bild 1). Man braucht nicht mehr zwischen den Icon-Ebenen hin- und herzuschalten, es sind ständig alle erreichbar. Das Fenster erlaubt den problemlosen Betrieb auf diversen Grafikkarten. Besonders interessant ist da die Farbgrafikkarte von MAXON, denn der Drafter beherrscht die Darstellung in 16 Farben. Das erleichtert die Arbeit natürlich ungemein, auch wenn die erforderlichen Geräte kostenaufwendig sind.

An die Arbeit

Die Arbeit mit dem Technobox Drafter gestaltet sich unproblematisch und ist stellenweise komfortabel. Gegenüber dem CAD/1 fällt die geringere Geschwindigkeit ins Auge. Das Fadenkreuz schleppt sich hinter der Mausbewegung her, wobei der Verlust sicher eine Folge der Flexibilität in Punkto Bildschirm ist. Einige Befehle wurden weggespart, so z.B. das Kopieren und Bewegen einzelner Objekte, jetzt läßt sich dies nur noch mit Bereichen bewerkstelligen. Diese Verluste lassen sich aber verschmerzen.

Was dem Drafter völlig fehlt, sind die diskettenfüllenden Zusätze wie Symbol-

bibliotheken. Die wird man sich erst einmal selbst erstellen müssen - lästig. Es fehlen ebenfalls die Software-Schnittstellen z.B. zum Einlesen von HPGL-Dateien oder zur Ausgabe im AutoCAD-Format DXF. Eine bequeme Möglichkeit, eigene Meßwerte oder dergleichen grafisch darzustellen, bestand in der ASCII-Schnittstelle. Die Daten konnten als Textdatei eingelesen werden und es entstand daraus eine Zeichnung. Leider ist auch diese der Einsparung zum Opfer gefallen.

Ich habe das abgebildete Beispiel gezeichnet und dabei sind mir einige Dinge aufgefallen. Während Grundlegendes wie Wahl des Papierformats, des Rasters, der Zeichnungseinheiten, des Maßstabes etc. routiniert gelöst sind und jedem Anwender gerecht werden dürften, gilt das für andere Details nicht.

Beispiel: Während des Aufziehens einer Strecke (das heißt: Startpunkt ist bereits festgelegt, Endpunkt soll plaziert werden) ist es wichtig, ständig über momentane Länge, Winkel der Strecke sowie über die Koordinaten des Endpunktes informiert zu werden. Der Drafter beschränkt sich jedoch auf eine Anzeige lediglich der X- und Y- Verschiebung bezogen auf den Startpunkt. Hat man Anzeige von Polarkoordinaten gewählt, dann sind es Länge und Winkel. Konsequenz: Vor dem Zeichnen der Strecke überlegen, was man braucht und dies entsprechend einstellen. Doch solche Verrenkungen sind nicht Stand der Software-Technik.

Für die Praxis

Lob verdient die Möglichkeit, während des Zeichnens - beispielsweise vor dem Setzen des Endpunktes der Strecke - den Zeichnungsausschnitt zu wechseln. Auch andere Parameter wie Fangen oder das Raster können geändert werden, ohne die Zeichnungsfunktion zu verlassen. Die Koordinateneingabe ist auch mittels Tastatur möglich. Fatalerweise hat das Handbuch an genau dieser Stelle einen Fehler und gibt ein falsches Beispiel, aber man hat ja Phantasie... Dabei brauchen für eine Strecke z.B. nicht alle vier Koordinaten eingegeben zu werden, sondern sie lassen sich Stück für Stück festlegen. Beispiel: Soll der Startpunkt bei X=0 liegen, gibt man 'x0' ein. Seine Y-Koordinate wird mit der Maus festgelegt. Die Länge der Strecke in Y-Richtung betrage 100, dann gibt man 'y100' ein, der Mausklick legt dann nur noch den X-Wert fest. Leider fehlt den Bochumern auch an dieser Stelle der Mut: Die Tastatureingabe z.B. auch von Winkeln oder Längen wäre ein großer Vorteil.

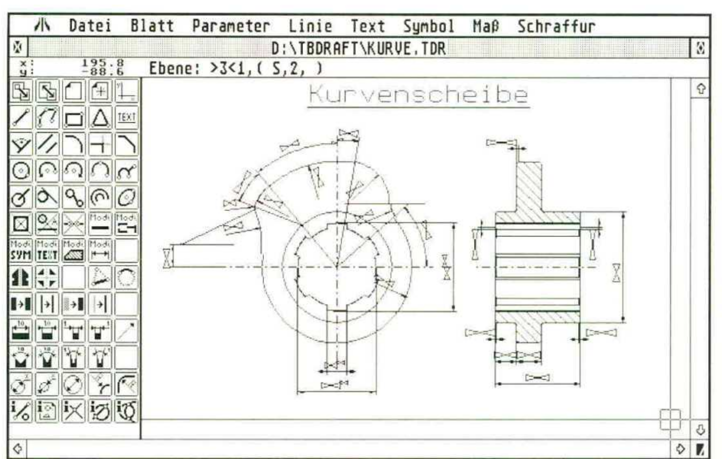


Bild 1: Die Arbeitsoberfläche Drafters



Bild 2: Das externe Ausgabeprogramm

In die Tastatureingaben lassen sich auch Menübefehle aufnehmen, beispielsweise die Wahl von Linienart und Stift. Die Befehle lassen sich auf den Funktionstasten speichern und bilden damit Quasi-Makros. Quasi deshalb, weil die Zeichenbefehle wie Kreis, Strecke etc. keine Tastenkürzel besitzen und folglich nicht 'programmiert' werden können.

Catch me

Eine der wichtigsten Funktionen bei jedem CAD-Programm ist die Fangfunktion, denn das oberste Gebot ist Präzision. Ist das Fangen aktiviert, sucht das Programm nach der Festlegung eines jeden Punktes zunächst in der Umgebung (Größe einstellbar), ob nicht vielleicht ein anderer, bereits gezeichneter Punkt existiert, der dann angesprungen wird. So beginnen dann mehrere Linien wirklich an einer Ecke. Hier macht der Drafter leider keine Unterscheidung zwischen der Art der zu suchenden Punkte: Er räumt zwar Endpunkten die höhere Priorität ein, doch ist in extremen Situationen nicht vorauszusehen, welcher Punkt das sein wird.

Ein besonderes Lob wiederum verdienen die Trimmfunktionen. Besonders herausragend: Das Trimmen von Kreisen. Leider ist das Verrunden von Kreisen nicht möglich, was mich sehr gefreut hätte - zumal, wenn man sich die Beispielzeichnung ansieht. Mit Kreisen und Kreisbögen hat nun mal ein jedes CAD-Programm so seine Schwierigkeiten. Wie bei vielen anderen auch, läßt sich mit dem Drafter ein Kreisbogen nicht tangential an einen schon bestehenden anlegen. Dieses Einpassen von Radien, das manuell/intuitiv mit der Kurvenschablone kein Problem ist - es wird mit dem Rechner zur Qual. Und ungenau obendrein.

Vollendet

Mit allen Zeichenfunktionen kommt man aber wunderbar und schnell zurecht - man muß vorher eben nur bedenken, was sie zu leisten imstande sind. Das Zeichnen selbst ist jedoch oft nur die Hälfte der Arbeit.

Bemaßen ist eine lästige und zeitaufwendige Angelegenheit. Hier glänzt der Technobox Drafter mit reichhaltigen Funktionen, die überdies noch durch entsprechende Parameter variiert werden können. Die Bemaßung der Beispielzeichnung war eine Arbeit weniger Minuten und wirklich ein Kinderspiel. Viel Erfahrung steckt in diesem Teil des Drafters. Doch hat die Sache einen Haken: Das Programm geht davon aus, daß Bemaßung Bemaßung ist und höchstens wieder gelöscht werden kann. Die Verlängerung beispielsweise der Radienbemaßung bis

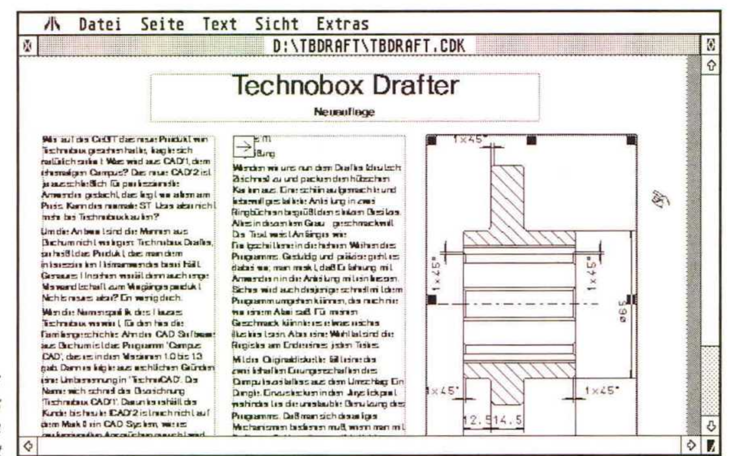


Bild 3: Drafter-Grafik als Metafile in Calamus importiert

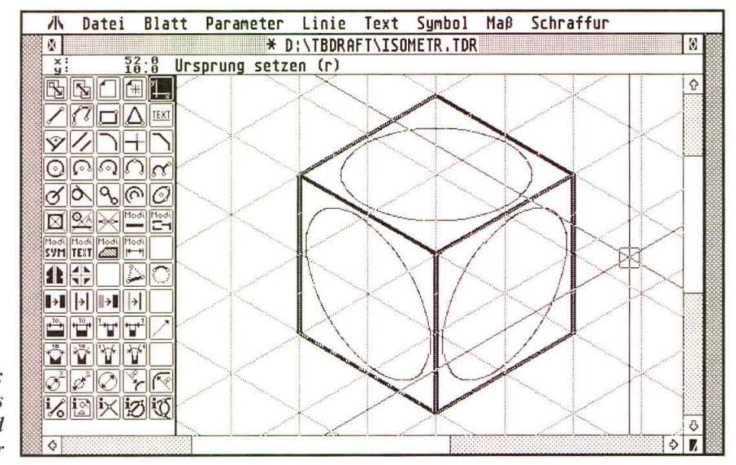


Bild 4: Perspektivisches Fadenkreuz und Raster

hin zum Mittelpunkt ist unmöglich. Lediglich der Text kann geändert werden. Auch sieht die DIN unterschiedliche Strichbreiten für Maßlinien und Text vor, was mit dem Drafter nicht realisierbar ist.

Ein ähnliches Problem entstand mit der Schraffur: Das Programm war nicht davon zu überzeugen, sie mit einem anderen als Stift 3 zu zeichnen. Immerhin kann man den Status 'Schraffur' auflösen und dann jeder einzelnen Linie den normgerechten Stift 1 zuordnen.

Einige Details machen Freude: Die helfende Undo-Funktion z.B. oder die Auto-Sichern-Funktion, die in einstellbaren Abständen die Zeichnung speichert und dann hilft, wenn keiner mehr hilft. Symbole werden in einer Auswahl über ihren Namen angesprochen. Alle Zeichnungsteile, die nicht auf der aktuellen Ebene liegen, können schattiert dargestellt werden. Wichtig für Großbildschirme: Die Dialoge (nicht die Warnmeldungen!) erscheinen dort, wo sich die Maus gerade befindet. In einer INF-Datei werden Farbeinstellungen und Pfade sowie Funktionstastentexte gespeichert, leider nicht Parameter wie Auto-Sichern, Schraffurbreite, Rasterabstand etc. Diese werden auch nicht mit den Zeichnungen gespeichert, müssen also jedes Mal wieder neu eingestellt werden. Nach jedem Befehl: 'Zeichnung ganz darstellen' hat das Fenster wieder die Proportionen der Zeichenfläche. Danach ist also ein Klick in die rechte obere Ecke fällig...

Ausgabe

Das Ausgabekonzept des Drafters ist identisch mit dem von CAD/1 bekannten. Ein externes Ausgabeprogramm verarbeitet spezielle PLT-Dateien, die das Programm anlegt. Das Druckprogramm wiederum bedient sich verschiedener Treiber, um die Geräte anzusprechen: Nadeldrucker, Laserdrucker, ATARI-Laser, PostScript und Plotter. Dabei sind verschiedene Gerätetypen individuell anzupassen. Besonders die Plotter-Anpassung bietet Möglichkeiten, auch das exotischste Gerät anzupassen. Das Programm selbst hat eine neue Oberfläche bekommen und ist nun komfortabel zu bedienen (siehe Bild 2). Leider besteht auf 24-Nadlern nur die Möglichkeit, in 180 mal 180 DPI auszugeben, also bei vielen Geräten nur der halben maximalen Qualität.

Das läßt sich aber mit einem Trick umgehen: Man exportiert die Zeichnung oder einen Teil daraus im GEM- Metafile-Format. Dafür gibt es wiederum ein Zusatzprogramm, das - wie das Druckprogramm auch - direkt aus dem Hauptprogramm aufgerufen werden kann. Das entstandene Metafile importiert man in Calamus, was problemlos vonstatten geht und auf komfortable Art und Weise die Kombination von Text und Zeichnung ermöglicht (siehe Bild 3). Dieses Dokument wird anschließend mit Calamus gedruckt und dann hat die Zeichnung die höchste Qualität. Doch GEM-bedingt leiden die Linientypen wie Strich-Punkt etc. darunter, denn die Abstände werden viel zu gering.

Perspektive oder nicht

Einer der wichtigsten Zusätze am Technobox Drafter ist das 'Perspektivenmodul'. Darunter verbergen sich Hilfsmittel zur Erstellung sog. axionometrischer Darstellungen. Das sind räumliche Darstellungen, die aber keinen Fluchtpunkt besitzen, sondern bei denen alle nach hinten laufenden Linien zueinander parallel sind, was die Arbeit sehr erleichtert. Dazu zählen die iso- und die dimetrische Darstellung wie die Kavaliersperspektive. Beim Drafter lassen sich die Winkel der X- und Y-Achse ebenfalls einstellen.

Unter dem Modul verbergen sich zwei Dinge: Zum einen bekommt das Fadenkreuz drei Fäden in Richtung der drei Achsen, zum andern verwandelt sich das Raster entsprechend (siehe Bild 4). Wer nun gedacht hat, die Koordinatenanzeige würde zur Anzeige in drei Achsen übergehen, statt Rechtecken würden Parallelogramme entstehen oder der Drafter würde Ansichten automatisch scheren, der wird enttäuscht sein. Das wäre zuviel verlangt von einem Programm dieser Preisklasse.

Aber einen Nebeneffekt hat dieser Programmteil doch gebracht: Der Drafter hat nun Ellipsen. Denn projizierte Kreise sind nun mal Ellipsen und die müssen ja jetzt dargestellt werden können. Leider zerfällt die Ellipse, nachdem sie gezeichnet wurde, in die Kreisbögen, durch die die Software sie intern darstellt. Eine Ellipse ist also nur in vielen kleinen Teilen wieder zu löschen.

Das Ende

Für wen ist der Drafter gedacht? Die Leute von Technobox wollen sich damit ihr Standbein auf dem klassischen ATARI-Markt erhalten. Doch mit 798.- DM ist ihr Programm nicht in der niedrigsten Preisklasse, wenngleich so etwas auf anderen Systemen mit Sicherheit weit teurer wäre. Auf dem ST aber muß es dem Vergleich mit anderen Produkten für diesen Preis standhalten. Das sind vor allem CADjA und CAD project. Laut Technobox wurde am Drafter alles das weggelassen, was nur der professionelle Zeichner benötigt, z.B. die Stücklistenerstellung. Es soll mit ihm ein reines Zeichenprogramm angeboten

werden, keine CAD-Software. Die Abspeckungskur, der CAD/1 unterzogen wurde, ist nicht überall von Vorteil. Natürlich will man sich mit dem Drafter in Bochum nicht das eigene Wasser abgraben: Das demnächst erscheinende CAD/2 soll ja echte Vorteile bieten.

Vorteile des Drafters liegen auch in seiner geringen Hardware-Anforderung. Ein 1040er und 1MB Speicher, das reicht ihm. Er läuft auch tapfer mit nur einem Diskettenlaufwerk. Wer für Darstellungen im Rohrleitungsbau oder in der Architektur auf Zeichenhilfen wie sie das Perspektivenmodul bietet, angewiesen ist und wem diese reichen, der wird mit dem Drafter sein Programm gefunden haben. Bei allen anderen Anwendungen bietet sich mit dem Produkt von Technobox ein solides und funktionables Werkzeug, um Zeichnungen zu Papier zu bringen. Ich persönlich vermisse die Ausstattung, die das Technobox CAD/1 bietet.

Ingo Brümmer

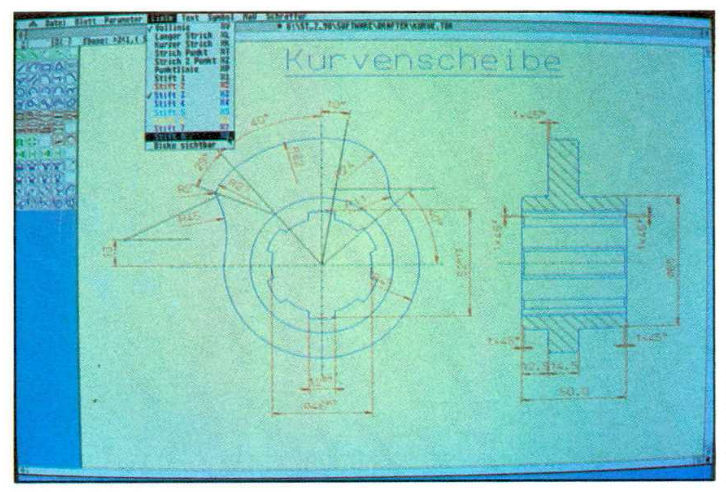


***Bild 5:* Drafter läuft auch mit der MAXON Grafikkarte zusammen.**

Literatur:

[1] CAD-Workstation ST. ST-Computer 6/89 S. 16 ff.

Heaven's Gate

ATARI ST

HARLEKIN

Inside heaven: reset-fester Editor, Diskmonitor (Dateien und Sektoren), Formatierprogramm, reset-feste Datenbank und Terminplaner, Taschenrechner, reset-feste RAM-Disk, Tastatur-Macro-Programmer, Kopierprogramm, luxuriöses Diskutility, einstellbarer Wecker und (Stopp-)Uhr, intelligenter Druckerspooler mit optionaler Ausgabe auf Disk, Bildschirmschoner, Systemzeichensatzeditor und -installer, Druckkonverter, ASCII-Tabelle mit Übernahme in GEM-Programme, Terminalprogramm, Tastatur-Reset (Warm- und Kaltstart), eigene System-Fileselectorbox, stark erweitertes Kontrollfeld, Quickmouse, RS232-Einstellung,... u.v.a.m..

1 MB und Monochrommonitor erforderlich

Harlekin ab Versand inkl. Porto u. Verpackung DM 136,50

Auslandsbestellungen nur gegen Vorauskasse. Prospekt anfordern.



DM 129,-

Unverbindlich empfohlener Verkaufspreis

MAXON Computer GmbH • Schwalbacher Str. 52 • 6236 Eschborn • Tel.: 06196/481811 • Fax: 06196/41885

Die Quelle allen Lichts

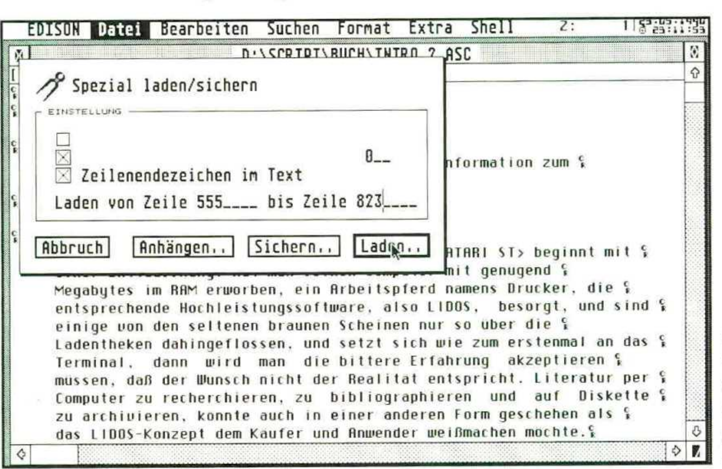
Schlaglichter auf die neue Editorengeneration am ST

Noch immer geistert der befremdliche Spruch eines Rezensenten durch die Fachpresse, jeder Mann pflanze in seinem Leben einen Baum, zeuge ein Kind und programmiere einen Editor. Auch die integrierte Programmier- und Entwicklungsumgebung Edison weckt mit dem direkten Namensbezug zum Erfinder der elektrischen Helligkeit Assoziationen zu menschlichen Glanztaten. Die Ironie, die in dem vorangestellten Zitat noch merklich mitschwingt, würde jedoch vor Edison verblassen. In meinem Überblick über die Leistungsmerkmale und -stärken dieses Fullscreen-Editors soll der Leser sowohl mit den Qualitäten von Edison als Programmer's Environment als auch unter dem Aspekt des Einsatzes als Textverarbeitung und für die Texterfassung beim Desktop Publishing bekanntgemacht werden. Die weitreichende Anwendungspalette enthält eine Vielzahl von Arbeitsschwerpunkten, die mit Edison in Angriff genommen werden können.

Ausgeliefert wird Edison auf einer Diskette, auf der sich die Programmdatei, mehrere Ladedateien für Jobs, Fonts und ein Resümee sowie ein Font- und ein Demo-Ordner befinden.

Man kopiert den Inhalt der Programmdiskette einfach in den vorbenannten Ordner einer Festplattenpartition, und auf geht die Arbeit. Wie die Pulldown-Menüs zeigen, ist Edison voll in GEM eingebunden. Dazu gibt es diverse Dialogboxen zur Einstellung von Parametern. Doch davon später mehr.

Ein Blick auf die Menüleiste beweist, daß wir es hier mit einem GEM-Programm zu tun haben. Nach Aussage des Handbuchautors leistet die GEM-Benutzeroberfläche eine effiziente Dialogführung und verringert die Einarbeitungszeit. Auch ohne könnte der Edison-Benutzer weite Teile des Programmes verstehen oder kurzfristig mit der vielberufenen Trial-and-Error-Technik erarbeiten. Eine Resource-Datei, wie sie bei GEM-Programmen standardmäßig dazugehört, entfällt.



Die Zange ist das treffende Icon für dieses Menü. Konvertierungsprobleme werden spezialbehandelt.

Die Edison.RSC hat man nämlich in das Programm integriert und dabei gutkalkuliert einen Geschwindigkeitsgewinn erzielt. Trotz der vieltalentierten Leistungsfähigkeit kommt Edison bytemäßig mit seinen rund 146K (plus den verschiedenen Ladedateien) im Zwergenwuchs dahermarschiert. Software-Bastler und Autoren ohne Massenspeicher können hiervon nur profitieren.

Beim Ladevorgang werden verschiedene Hilfsdateien aufgerufen. Mit der Datei *Edison.FNT* kann ein GEM-Zeichensatz eingeladen werden; *Edison.INF* beinhaltet alle Einstellungen aus den Dialogen und der Menüleiste; die LIN-Datei enthält Informationen zur Linealeinstellung einer Textdatei und trägt deren Namen mit dem Suffix *.LIN*. Schließlich gibt es eine wenig stabile Resümee-Datei (Fehler: Inkompatibilität mit dem Festplattenbeschleuniger COPYFIX.PRG von Hard&Soft), die den Inhalt einer Textdatei bei Stand des Verlassens archiviert und automatisch beim Programmstart wieder einlädt. Das Resümee ist beim Startvorgang per Shift-Taste zu unterdrücken. Selbstverständlich kann die automatische Sicherung der Hilfsdateien *.INF*, *.LIN* und *.RSM* per Dialogmenü unterdrückt werden. Es können bis zu sechs Dateifenster gleichzeitig geöffnet sein. Interessant ist der Menüpunkt *Spezial* ... des Dateiauswahlmenüs. Dieser Menüpunkt ist für besonders hartnäckige Datenimporte mit Spezialformaten zuständig. Edison prüft alle Formatangaben, die nicht nach dem ASCII-Standard ausgelegt sind, und lädt anschließend solche Dateien.

Wobei wir beim Wichtigsten von Edison angelangt wären: den Dialogen. Die Dialoge helfen dem Benutzer, sich schnell im Editor zurechtzufinden und mit Edison eine den privaten Zwecken entsprechende Arbeitsumgebung aufzubauen. Eine sehr nützliche Funktion im Menü Bearbeiten

ermöglicht das Widerrufen. Die letzte Textmanipulation wird rückgängig gemacht, und es erscheint der Text vor seiner Veränderung. In Kombination mit der UNDO-Funktion arbeitet das Widerrufen blockorientiert. Man bedient sich beider Funktionen per Schalter.

Neben den bei vielen Editoren üblichen und unverzichtbaren Blockoptionen wie Ausschneiden, Kopieren und Einsetzen bekommt man hier einige zusätzliche Features präsentiert, die Edisons Textverarbeitungsqualitäten demonstrieren. Meines Wissens hat allein SCRIPT ähnlich ausgetüftelte Blockfunktionen. Sehr praktisch ist das Verschieben von Textblöcken nach links und rechts, aber auch das Reformatieren des Textblockes linksbündig mit Flatterrand. Rechtsbündig ohne Flatterrand mit ausgezeichneter Silbentrennung würde als Wunsch aller Textverarbeiter nachdrücklich an Edisons Adresse gerichtet werden. Ohne diese Fähigkeiten bleibt Edison der flexible Editor, der es zu sein beansprucht, obwohl vieles in Richtung Textverarbeitung weist. Deshalb gibt es hier nur das Angebot <*Reformatieren*> statt <*Formatieren*>. Wie sagt man doch? Was noch nicht ist, kann ja noch werden. Und bei Edison in der Version 1.0 sind bei diesem Leistungsumfang noch Entwicklungsschritte drin. Es gibt zwar einen automatischen Zeilenumbruch, der allerdings nicht mit einer Trennungsdatei verknüpft ist. Der Zeilenumbruch richtet sich nach der Linealeinstellung des rechten Randes und bricht ausschließlich ganze Wörter um. Auf Wunsch läßt sich jeder Text automatisch einrücken. Bei der Texterfassung arbeitet man im Fließtextmodus. Für manche Aufgaben, wie das Erfassen von Datensätzen, ist die Kennzeichnung der Leerräume zwischen den einzelnen Zeichen mit Tabulator- und Wagenrücklaufzeichen (CR/LF) aufgefüllt. Auf dem Bildschirm erscheinen die Tabulatoren als dicker Punkt. Das Lineal arbeitet mit den sogenannten 'echten' Tabulatoren, d.h. der Zwischenraum zwischen zwei Spalten wird als ein Zeichen ausgefüllt. Erwähnenswert ist auch, daß alle Zeichen des ASCII-Zeichensatzes als Tabulatoren Verwendung finden können.

Eine Speicherplatzanzeige informiert Sie, wann das RAM zum überlaufen kommt. Dann wird eine <*Garbage Collection*> fällig. Der Speicher wird vom Datenmüll gereinigt, und es erfolgt seine Reorganisation.

Das Menü *Suchen* enthält im Menüpunkt *Block übernehmen* eine sehr effektive Suchbedingung. Immer vor dem Hintergrund, daß Programmierer wie Textbearbeiter von Edison unterstützt werden, lassen sich ganze Textblöcke als Suchkriterien definieren. Hiermit kann man nach sich eventuell wiederholenden Sätzen im Text-File oder nach einzelnen Befehlsketten im Programmcode suchen. Daneben gibt es auch die heute üblichen Suchoptionen wie *weiter suchen* oder *weiter ersetzen*.

Format
- Fenster blättern ^W
- Fenster stapeln ^,
- Fenster Übersicht ^.
- ✓ Lineal sichtbar
- ✓ Unsichtbare Zeichen
- ✓ Tabulatoren expandiert
- ✓ Autom. Einrücken
- ✓ Autom. Zeilenumbruch
- ✓ Auffüllen mit Tabs
- ✓ Einfügemodus ^Ins

Bearbeiten
- Widerrufen Undo
- Ausschneiden ^X
- Kopieren ^C
- Einsetzen ^V
- Alles auswählen ^H
- Links schieben ^[
- Rechts schieben ^]
- Reformatieren ^R
- Chr->Hex wandeln ^#
- Groß/Klein ^~
- Alles groß ^+
- Alles klein ^-
- Kapitalisieren ^'

Datei
- Neu ^N
- Öffnen.. ^O
- Zuladen..
- Spezial..
- Sichern ^S
- Sichern unter.. ^U
- mit Backup
- Zweiter Pfad..
- Druckformat..
- Drucken.. ^D
- Schließen ^ß
- Ende ^Q

Suchen
- Suchen / Ersetzen.. ^F
- Block übernehmen ^Y
- Weiter suchen ^A
- Weiter ersetzen ^E
- Tabs in Leerzeichen
- Leerzeichen in Tabs
- Zeilenenden säubern
- Gehe nach.. ^G
- Letzte Position Esc
- Fehlersuche Help

Extra
- Speicher 1102 KB
- Voreinstellungen..
- Zahlenumwandlung.. ^Z
- Lineal einstellen.. ^L
- Klammern einstellen..
- Textinformationen.. ^I
- Zeichentabelle.. ^T
- Makro aufnehmen ^M
- Position merken ^0-9
- Sortieren.. ^<

Shell
- Programm starten.. ^P
- Dateinamen suchen..
- JOB-Datei zeigen ^J
- JOB-Datei übernehmen ^Ü
- Compile Top Window F1
- Link Top Window PRG F2
- START Top Window.PRG F3
- Link Top Window TOS F4
- START Top Window.TOS F5
- RSC aufrufen F6

Das ausufernde Menüangebot mit Shortcuts

Mir ist allerdings unverständlich, warum das Suchen-/Ersetzen-Menü nicht mit Wildcards funktioniert, obwohl diese zur Dateisuche, z.B von der Shell, benutzt werden. Insbesondere beim Programmieren, wo nicht wortweise, sondern nach Zeichenketten mit teilweise unbekannten Elementen gesucht wird, würde eine zusätzliche Ausstattung des Suchmenüs mit Suchquantoren einen großen Vorteil bedeuten.

Der Menüpunkt *Zeilenende säubern* reinigt den Zeilenschluß von überhängenden Tabulatoren mit dem Effekt, daß einige Kilobytes der Speicherverwaltung gutgeschrieben werden. Praktisch für alle Textverarbeiter ist die Funktion der ESC-Taste. Sie organisiert das Zurückspringen auf die letzten zehn Positionen, an denen etwas eingegeben oder verändert worden ist. Viele, aber weniger interessante Funktionen, die Sie hier vermissen mögen, und die die Arbeit mit einem Text-Editor so wertvoll machen, kann Edison allemal.

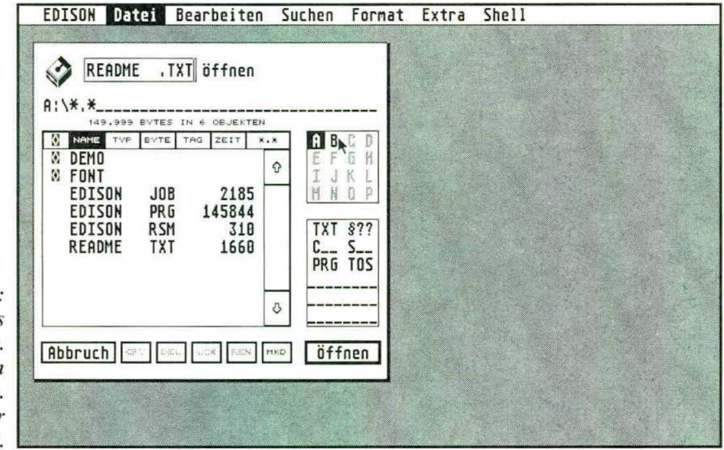
Bei der Masse an Funktionen und Optionen, Parametereinstellungen und Bedienungsvorschriften muß man schon die übervolle Angebotspalette von Edison sichten und sortieren, um schließlich einen Eindruck von seiner Leistungsfähigkeit zu haben.

Lassen Sie uns eine weitere Etappe unseres Software-Überblicks absolvieren. Von den Menüs gelangen wir überleitend zu den Dialogen. Die dialogische Benutzerführung von Edison ist ein weiters Plus. Man hat Dialogboxen konzipiert, die vom gewohnten GEM abweichen und ein wenig an Macintosh-Dialogboxen erinnern.

Wegen der manuellen Einstellbarkeit aller Menüunterpunkte würden das einfache Abhaken eines Befehls oder einfache Parametereingaben nicht mehr ausreichen. Edison ermöglicht differenziertere Dialoge. Zur Visualisierung des jeweiligen Programmpunktes wurden menüintern sprechende Icons und erklärende Überschriften gewählt. Die Ampel mit der berühmten Drei-Phasen-Schaltung Rot, Gelb und Grün signalisiert dem Anwender beispielsweise, wann gestoppt, aufgepaßt oder losgelegt werden kann. Zur Regelung der Menüfunktionen werden Schalter und Radioknöpfe eingesetzt. Durch Schalter sind mehrere gleichrangige Menüpunkte zu aktivieren oder deaktivieren, wogegen mit Radioknöpfen nur eine von mehreren angebotenen Funktionen bereitgestellt werden kann. Und schließlich fallen bei einigen Menüs noch Text- und Parametereingaben an, die durch eine typische GEM-Eingabemaske erstellt werden dürfen.



Das Suchen/Ersetzen-Menü zeigt deutlich Textverarbeitungsmerkmale.



Zweckarchitektur: Edisons Fileselector. Unschön anzuschauen. Praktisch bei der Arbeit.

Wird ein Untermenü aktiviert, erscheint es an der aktuellen Mausposition auf dem Bildschirm. Besonders bei Großbildschirmen dürften damit die vielen langen, umständlichen und Zeit kostenden Mauswege der Vergangenheit angehören.

Sicherlich würde die Einzeldarstellung der vielen hilfreichen Dialogpunkte zu weit führen. Das komfortable Arbeiten mit Edison beginnt quasi von der ersten Sekunde an mit dem Einladen einer Textdatei. Nicht nur, daß man (wie schon besprochen) Texte mit exotischen Zeichenformaten bearbeiten kann, Edison erlaubt auch die Voreinstellung von Zeilenlänge, Absätzen und Textblöcken im eingeschleusten Text-File. Man bekommt hier die luxuriös zu nennende Gelegenheit, den Textimport vor dem Ladebeginn auszusteuern. Das kann bei der Zusammenfassung von mehreren Dateien zu einer Listen-Datei sehr von Vorteil sein. Auch bezüglich der Importierung von Datensätzen ist hierin eine nützliche Anwendungsfunktion zu sehen. Zeilenweise numerierte Datensätze können während des Ladevorgangs vorsortiert werden. Vorteil: Dem Benutzer bleibt die Desorganisation von nicht benötigten Datensätzen erspart. Zu allem Überfluß funktioniert die flexible Steuerung des Datenmaterials auch beim Sichern. Der Programmierer bekommt hier Gelegenheit, Programmbibliotheken oder Auszüge aus einem Quelltext beim Lade- und Speichervorgang projektmäßig in einzelne Textmodule zu gliedern.

Edison benutzt eine in seiner Funktionsvielfalt sehr mächtige Dateiauswahlbox. Sie enthält alle Optionen, wie Formatieren, Datei löschen und umbennen, die man heute erwarten darf.

Wo soviel mit Texten gearbeitet wird und von Texten die Rede ist, darf eine leistungsstarke Drucker-Schnittstelle nicht fehlen. Der Druckertreiber ist standardmäßig für Epson-Nadler ausgelegt. Besitzt man einen Hochleistungs-Printer anderen Typs, fallen einige Manipulationen zur Anpassung des Druckertreibers an. Sofern Ihnen kein Drucker von exotischer Herkunft Fußangeln legt, reicht der mitgelieferte Epson-Druckertreiber weitlich aus, Ihrem Schnell-Picker formschöne Schrift auf weißem Hintergrund zu entlocken. Insbesondere im Hinblick auf den Einsatz eines Laserdruckers muß ein geeigneter Druckertreiber aufgebaut werden, wenn auch das Edison-Handbuch sich über diese Anwendung ausschweigt. Dennoch, der Druckertreiber ist hervorragend dokumentiert. Mit ein wenig Erfahrung bei der Manipulation der Druckercodes (Font-Größen, Zeichenmodi) werden hier Möglichkeiten zur Edierung des individuellen Druck-Layouts präsentiert.

Das Menü *Druckformat* zeigt deutlich Edisons Tendenz zur Textverarbeitung. Es ist sehr aufwendig mit Funktionen bestückt, die eine produktive Textgestaltung überzeugend wirken lassen. Immer wieder schön ist es, Manuskripte mit blattweisen Überschriften zu versehen oder Kommentare an das Seitenende zu hängen. Edison bietet jeweils eine einzeilige Beschriftung der Kopf- und Fußzeile. Zeilenränder und Zeilenabstände lassen sich ebenso einstellen wie verschiedene Papierformate oder Schrifttypen. Es braucht auch kein Nachteil zu sein, daß die Druckerschnittstelle auf die Portierung von ASCII-Zeichen ausgelegt ist. Man hat also die Wahl zwischen den drei hinlänglich bekannten Schriftarten Pica, Elite und Micro sowie die Umschaltoption zwischen NLQ- und Draft-Modus. Variable Zeichenabstände, also Proportionalschriften, bereiten Edison kleine Schwierigkeiten. Man hat deswegen den Radioknopf speziell zur Berechnung der Zeichenabstände eingerichtet.

Besser als so manche Dateiverwaltung leistet Edison die Sortierung von Textpassagen. Mit dieser Funktion lassen sich Exportdateien aus Datenbanken bearbeiten. Der Sortiervorgang organisiert sich nach folgendem Prinzip. In der Referenzzeile blockt man einen Prototypdatensatz, legt den datensatzspezifischen Zeilentrenner fest und definiert anschließend das Kriterium, nach welchem sortiert werden soll. Bis zu drei Kriterien werden bei der Sortierung unterstützt. Die UNDO-Funktion stellt den anfänglichen Zustand wieder her, wenn die Sortierung nicht den Erfordernissen entspricht. Das Sortieren der privaten CD-Sammlung stellt mit dieser Funktion, wie das Handbuch betont, kein Hindernis dar.

Es war soviel von Edisons Textverarbeitungsqualitäten die Rede. Nun, das ist die eine Seite des Editors. Die andere Seite, das Volk der Programmierer (wechselweise Staatsbürgerschaften sind erwünscht), wird mit einigen aufregenden Features unterstützt. Mit der Zahlenumwandlung besitzt Edison ein Dialogfeld, das einem das Be- und Umrechnen von Zahlencodes abnimmt. Praktisch jedes darzustellende Zeichen ist in die bekannten Zahlensysteme Octal, Binär, Hexadezimal umzuwandeln.

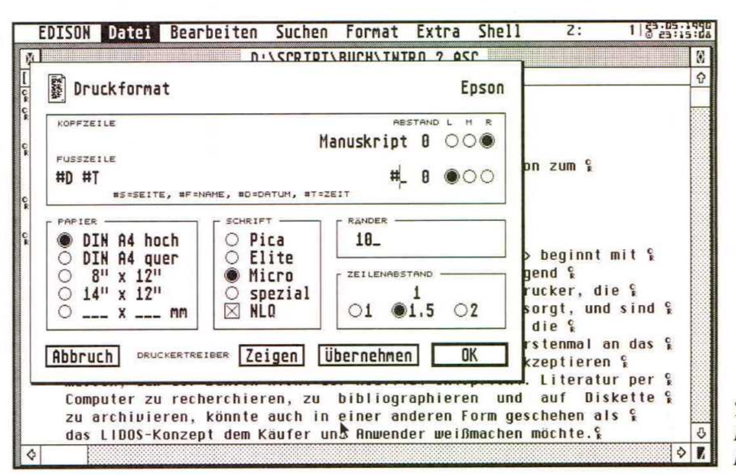
Bevor es an die Zahlenumwandlung geht, baut man bekanntlich den Quellcode unter Einhaltung der Syntaxkonventionen einer Progammiersprache auf.

Die Eingabe von Befehlen und Schleifen oder besonders häufig wiederkehrenden Ausdrücken kostet in der Regel Zeit, die man für andere Dinge gebrauchen könnte. Edisons Klammerblock erlaubt die Definition von bis zu zehn Klammerausdrücken in einer Programmierumge-

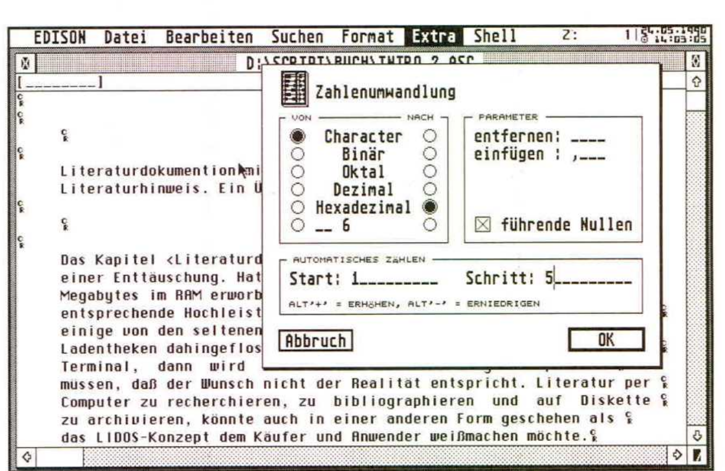
bung. Der jeweilige Klammerausdruck wird als Block behandelt. Einfache wie kompliziertere Klammerschachtelungen können leicht auf Klammerfehler getestet werden. Allerdings werden jeweils nur Klammerausdrücke auf einer Stufe, also *()* oder *()*, geprüft. Per Doppelklick auf das erste Element von Schleifen, Kommentaren, Strings etc. markiert man den Gesamtblock. Klappt diese Funktion nicht, muß irgendwo ein Klammerzeichen zur Kennzeichnung des Blockendes fehlen.

Edisons Etikette: Editor mit integrierter Entwicklungsumgebung, signalisiert, daß man sich für die Einbindung und Kontrolle von Programmierabläufen etwas Besonderes hat einfallen lassen. Mit der Belegung entsprechenden Job-Dateien lassen sich ganze Compiler- und Link-Vorgänge steuern. Federführend für die Definition und die Handlungen einzelner Job-Dateien zeigt sich das Shell-Menü. Bis zu vierzig Jobs können in einer Datei definiert sein. Beim Edison-Start wird automatisch der *Edison.JOB* installiert. Eine Befehlsliste hilft, den Programmjob entsprechend der gewählten Programmierumgebung aufzubauen. Jeder der maximal vierzig in einer Job-Datei definierten Einträge oder Befehle kann einzeln angesteuert werden. Compiler-Läufe können also präzise per Funktionstasten F1-F10 (und in Kombination mit Shift) für die ersten zwanzig Befehlsaufraufe kontrolliert werden. Die restlichen zwanzig werden jeweils per Mausklick im Menü gestartet. Als Beispiel zum Aufbau einer Job-Datei liefert man eine Job-Anpassung des TURBO-C-Compilers mit, die allerdings entsprechend der hauseigenen Hardware-Bedingungen umdefiniert werden muß.

Von Shortscuts, den tastaturgesteuerten Funktions- und Menüaufrufen, war schon die Rede. Wo sehr ausgiebig von Software-Ergonomie und Arbeitserleichterung die Rede ist, dürfen Makros nicht fehlen. Makros fassen ganze Befehlssequenzen in einem Ausdruck zusammen und sind durch eine festdefinierte Tastenkombination zu aktivieren. Edison verwendet einen Ereignisrekorder. Wie bei einem Tonband können mit dem Ereignisrekorder homogene Funktionsabläufe aufgenommen und später wieder abgespult werden. Ob es das Aktivieren einer Job-Datei ist oder die Vorauswahl bestimmter Blockfunktionen, die Festlegung von Klammerausdrücken oder wiederkehrenden Prozeduren der Zahlenumwandlung, wirklich alles kann als Makro erfaßt werden. Man braucht es Edison nur vorzuexerzieren, was Inhalt einer Makroaufnahme sein soll, und einen Tastaturcode festzulegen, unter dem das Makro



Text- und Listendruck mit Komfort



Der elektronische Abakus als Zahlenumwandler

anzuwählen ist. Mit den Buchstaben Klein- und-Groß-A-Z inklusive der Shift-Taste sind maximal zweiundfünfzig Makrodefinitionen erlaubt. Außer der Fileselectorbox und dem Menü *Allgemeine Einstellungen* ist quasi das ganze umfangreiche Befehls- und Funktionsangebot von Edison als Makrodefinition erlaubt. Selbstverständlich kann man mit der Makroverwaltung auch ein Sortiment von Textfloskeln zusammenstellen. Das rekursive Aufrufen der Makros, beispielsweise der Suchen-/Ersetzen-Funktion oder Wiederholung von Sortiervorgängen, wird ebenfalls unterstützt. Nicht nur in der Handhabung der Makros über Tastaturkürzel, sondern auch bei der Definintion der Makrobedingungen beweist Edison sein enormes Talent.

Hier konnten nur einige Hauptpunkte angeschnitten werden, die in Zukunft Programmierern wie Textbearbeitern oder Desktop-Publishern das Arbeitsleben unbedingt erleichtern. Nicht enden darf ich, ohne auf das übersichtlich informierende und sachlich aufgebaute Handbuch zu sprechen zu kommen. Der Handbuchautor klopft Edison für meinen Geschmack zu sehr auf die Schultern. Vermutlich soll das so sein, um die Anwenderschaft von der Qualität und der Ausgereiftheit der einzelnen Programmdetails zu überzeugen. Wer dann die Arbeit mit dem Editor Edison aufnimmt, mag auch verzeihen können, daß Eigenlob stinkt. Edison jedenfalls wird als integrierte Programmier- und Entwicklungsumgebung viele Freunde finden. Die altbewährte und betagte Konkurrenz wird, auch ohne Hardwaredoping, noch um Längen geschlagen. Ausschlaggebend für das positive Resümee sind die vielen und gutdurchdachten Features, die Edison in sich vereinigt. Wer schon Software am TT fahren kann, ist gut dran, denn er darf hardwaremäßig unbeeinträchtigt das selbsternannte Software-Genie zu Höchstleistungen hochpeitschen.

Mit Edison tut man einen direkten Schritt in die Zukunft des editorunterstützten Publizierens, Archivierens und Programmierens. Selbstredend wäre noch zu ergänzen, daß Edison bei der aufgebotenen Funktionsvielfalt ein sehr schnelles Software-Werkzeug ist, das nicht nur eine hohe Scroll-Geschwindigkeit auf die Waage bringt, sondern auch sehr ökonomisch mit der Belegung von Speicherplatz umgeht. Was würde wohl Alva Edison dazu gesagt haben? Software-Käufer werden mit DM 169.- zufriedene Gesichter machen. Schließlich ist Edison keine Software für Leuchten.

Ralf Blittkowsky

Bezugsadresse:

KNISS Soft Adalbertstr. 44 5100 Aachen
Tel. 0241/24252

Einkaufsführer

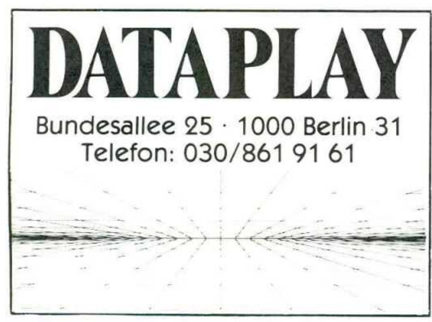
Hier finden Sie Ihren Atari Fachhändler

1000 Berlin





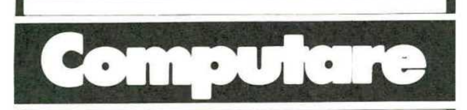






1000 Berlin





Keithstr. 18-20 • 1000 Berlin 30
030/21 390 21
186 346 com d

Ihr Atari-Händler in Berlin
COMPUTERSHOP Radtke u. Kögel
Riesen Software Angebot
Fürbringerstr. 26 · 1000 Berlin 61
Tel. (030) 6 91 46 29 · BTX (030) 6 91 76 66





1000 Berlin



PUBLIC DOMAIN SERVICE

Sämtliche verfügbaren ST COMPUTER - PD`s. vorrätig und weitere Serien für den ST, z.B. PD-Pool und ST-Vision.

1000 Berlin 65 * Pankstr. 42
Tel.: 030/465 70 28

SERVICE STATIONEN
Auch hier alle PD`s vorrätig!
1/44, Lahnstrasse 94
1/20, Schönwalder Str. 65





2000 Hamburg

Planen • Beraten • Realisieren



ATARI Fachhändler

Münsterstraße 9, D-2000 Hamburg 54
Telefon 040/56 19 09-0
Telefax 040/56 19 09-80

2000 Hamburg



RADIX Bürotechnik

Heinrich-Barth-Straße 13
2000 Hamburg 13
Telefon (0 40) 44 16 95

NEU: Software Shop



DTP u. Werbeservice

Soft- und Hardwareversand

Viola Jaap
Berner Heerweg 512
2000 Hamburg 72
Telefon (0 40) 6 44 06 68

2000 Norderstedt



Ulzburger Str. 2
2000 Norderstedt
Tel. 0 40 / 5 27 30 47

2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt

Heiligengeiststr. 20, 2120 Lüneburg
Tel. 0 41 31 / 4 61 22, Btx 40 24 22
Mo.-Fr. 9⁰⁰-18⁰⁰ und Sa. 9⁰⁰-13⁰⁰

2210 Itzehoe



Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90 / 91

2300 Kiel



Die Welt der Computer
Dreiecksplatz Nr. 7
2300 Kiel 1 · ☎ 04 31 / 56 70 42

2800 Bremen

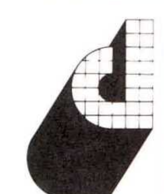


Faulenstraße 48—52
2800 Bremen 1
Telefon (04 21) 17 05 77

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler
Markstr. 52
2940 Wilhelmshaven
Telefon 0 44 21 - 2 61 45

3000 Hannover



DATALOGIC
COMPUTERSYSTEME

ATARI ST-	BERATUNG
COMPUTER	SERVICE
HARDWARE	VERKAUF
SOFTWARE	

CALENBERGER STR. 26
3000 HANNOVER 1
TEL.: 0511 - 32 64 89

COM DATA

Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36

3400 Göttingen

Büroeinrichtungs-Zentrum
Wiederholdt

3400 Göttingen-Weende
Wagenstieg 14 – Tel. 05 51 / 38 57-0

3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler

Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (05 61) 70 00 00

4000 Düsseldorf

BERNSHAUS GmbH
Bürotechnik – Bürobedarf

Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81



4000 Düsseldorf
Erkrather Straße 169-179
Tel. (0211) 73 62 14 • Fax 73 31 737

Hard und Software
Werner Wohlfahrtstätter

Atari	**Ladenlokal**
Public Domain	Irenenstraße 76c
Atari Spiele	4000 Düsseldorf-Unterrath
Atari Anwender	Telefon (02 11) 42 98 76

HOCO
EDV ANLAGEN GMBH
Ellerstraße 155
4000 Düsseldorf 1
Telefon 02 11 / 78 52 13

4010 Hilden

Beachten Sie
unsere Anzeige
in diesem Heft!



Computer · Drucker · Software · Bücher · Service
Gustav-Mahler-Straße 42-44
Tel. (0 21 03) 3 18 80 + 4 12 26

4150 Krefeld



DTP-CENTER

MATRIX *MatScreen*

ROLAND *Schnittplotter*

DATACOPY *Scanner*

Digitizer

C.O.P Computer-Service GmbH
Tannenstr. 103
4150 Krefeld
☎ (02151) 77 30 41
FAX (02151) 77 05 86

4150 Krefeld



4150 Krefeld
Ostwall 138
Tel. (0 21 51) 15 03 • Fax 69 409

4200 Oberhausen

Redakteur · Spectre 128



für TOS, DOS, UNIX, MAC, AMIGA und andere Systeme
by COMPUTER MAI

dBMAN-Vertretung für NRW und BENELUX
ISYS-COMPUTER GbR
Tel.: 02 08 / 65 50 31 · Telefax: 02 08 / 65 09 81
Max-Eyth-Straße 47 · 4200 Oberhausen 11

4300 Essen

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Limbecker Platz 4300 Essen 1
Tel.: (02 01) 17 63 99



4300 Essen
Limbecker Straße 12-16
Tel. (02 01) 23 35 15 · Fax 23 38 34

4320 Hattingen



4422 Ahaus

ATARI · Epson · Fujitsu Molecular · NCR · Tandon · Schneider · Star

OCB

OCB-Computershop Wallstraße 3 4422 Ahaus Tel. 0 25 61/50 21	OCB-Hard- und Software Wessumerstraße 49 4422 Ahaus Tel. 0 25 61/50 21

4430 Steinfurt

CBS GmbH
COMPUTERSYSTEME
Tecklenburger Str. 27
4430 Steinfurt-Burgsteinfurt
☎ 02551/2555

4500 Osnabrück

Heinicke-Electronic
Meller Str. 43 • 4500 Osnabrück
Fax (0541) 58 66 14
Telefon (0541) 58 66 46
Wir liefern Micro-Computer seit 1978

4520 Melle

C B S GmbH
COMPUTERSYSTEME
4430 Steinfurt Tel. 02551/2555
Haferstraße 25 4520 Melle
Tel.: 05422/44788

4600 Dortmund

Elektronik
Computer
Fachliteratur
ATARI-System-Fachhändler
4600 Dortmund 1, Güntherstraße 75, Tel. (02 31) 57 22 84



city-elektronik

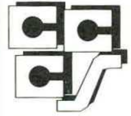
ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Kampstraße 1 · 4600 Dortmund
Telefon (02 31) 5 43 91

4600 Dortmund

cc Computer Studio GmbH



Atari-Systemfachhändler
PCs von Tandy Schneider Peacock
Drucker von Star Brother NEC
Elisabethstr. 5
4600 Dortmund 1
Tel. 0231/528184 Tx 822631 cccsd Fax 0231/528131

4650 Gelsenkirchen-Horst

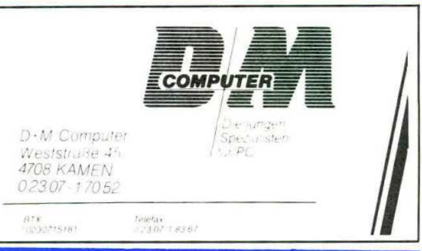
MENTIS GmbH
Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel
Poststraße 15 · 4650 Gelsenkirchen-Horst
Telefon (02 09) 5 25 72

4650 Gelsenkirchen



Fest- und Wechselplatten (SCSI)
Großmonitore, Scanner, Software
ComputerSysteme und Anwendungen
Hüttenstraße 56 – 4650 Gelsenkirchen
Tel.: (0209) 203420 FAX. (0209) 271584

4708 Kamen



4712 Werne

Vogler & Trümper
Hard- und Software



Lünener Straße 14
4712 Werne
Tel. (0 23 89) 5 14 95

4800 Bielefeld

hardware
software
organisation
service
CSF

CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106 – 108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

Bei uns werben bringt GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

5000 Köln

BÜRO MASCHINEN



AM RUDOLFPLATZ GmbH
5000 KÖLN 1
RICHARD-WAGNER-STR. 39
TEL. (02 21) 21 91 71

5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5210 Troisdorf

LOGITEAM
Computerhandelsgesellschaft mbH
Kölner Straße 132
5210 Troisdorf
Tel. (0 22 41) 7 18 97
FAX (0 22 41) 7 58 58



5253 Lindlar



5300 Bonn



5414 Vallendar



ATARI-Systemfachhändler für Mayen-Koblenz
Ihr autorisierter Fachhändler für GTC- Personalcomputer, Star, Epson und NEC

Wir schreiben BERATUNG und SERVICE groß!
Zentrale: 5414 Vallendar, Rheinstr. 117, TEL. 0261/61727
5419 Dierdorf, Hauptstraße 50
5500 Trier, Ehrangerstr. 31

5430 Montabaur



5500 Trier



Güterstraße 82 · 5500 Trier
☎ 06 51 / 20 97 10
Fordern Sie unsere Zubehör-Liste an!

5600 Wuppertal



KIPDORF 22 • 5600 WUPPERTAL 1 • TEL 0202 45 32 33

HARDWARE • SOFTWARE • ZUBEHÖR • SERVICE • SCHULUNGEN



MEGABYTE
Computer Vertriebs GmbH

Friedrich-Engels-Allee 162
5600 Wuppertal 2 (Barmen)
Telefon (02 02) 8 19 17

5630 Remscheid

COM SOFT
Nordstraße 57 · 5630 Remscheid
Telefon (0 21 91) 2 10 33

5650 Solingen

MegaTeam
Computer-Vertriebs-OHG
Kölbach – Finke
Hardware – Software – Zubehör – Service
Rathausstraße 1–3 · 5650 Solingen 1
Telefon (02 12) 4 58 88 · Fax (02 12) 4 73 99

5800 Hagen



Vertragshändler Axel Böckem
Computer + Textsysteme
Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Telefon (0 23 31) 7 34 90

5860 Iserlohn



5900 Siegen



Siegen · Weidenauer Str. 72 · ☎ 02 71/7 34 95

6000 Frankfurt

WAIZENEGGER
Büroeinrichtungen
Kaiserstraße 41
6000 Frankfurt/Main
Tel. (0 69) 2 73 06 - 0

Bei uns werben bringt

GEWINN

Sprechen Sie mit uns.
Heim Verlag ☎ 0 61 51 / 5 60 57

BUF

Eickmann Computer

Der Atari-System-Fachhändler !

z.B.: Festplatten von 30 MB bis 110 MB für Atari ST und Mega ST, Zusatzaufrüstungen für Ihre Festplatten bis 110 MB, Umrüstung Ihres SM 124 in einen EM 124 Multisync für alle Auflösungen, Slotkit für PC 1...

besuchen Sie unser Fachgeschäft:
In der Römerstadt 249
6000 Frankfurt 90-Praunheim
Telefon (069) 763409

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon (0 61 51) 5 60 57

6108 Weiterstadt



6123 Bad König



! Vor(Ver)führungen !

CALAMUS	OUTLINE ART	ARABESQUE
Repro Studio ST	PEGASUS +	DRAFTER CAD
TmS Graphics	TmS CRANACH	TmS Vektor
Sherlook OCR	SIGNUM ! 2	SCRIPT
ADIMENS PLUS	BSS Plus	Reprok
Eickmann Turbo 16 MHz		PC SPEED 1.4
PANASONIC Scanner 506 U	LACOM Wechselplatte	

Speichererweiterung 2 MB: 598.– / 4 MB: 998.–
Laufwerk 5 1/4" 40/80 Track: 298.–
Hard und Software aller Hersteller

6200 Wiesbaden





COMPUTERCENTER
DER SYNTHESIZER • STUDIO JACOB GMBH
Computer, Software, Zubehör
MIDI - Equipment, eigener Service
6200 WIESBADEN, MAINZER STR. 137
Tel. (06121) 71 94 90 BBS: (06121) 70 17 39

6236 Eschborn



6240 Königstein

KFC
COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6250 Limburg



DTP–Center • Salzgasse 6 • 6250 Limburg
☎ 06431/5004–0 • FAX 06431/5004–10

6300 Gießen



6400 Fulda

Schneider ATARI Commodore
WEINRICH
BÜRO · ORGANISATION
Ronsbachstraße 32 · 6400 Fulda
Telefon (06 61) 4 92 - 0

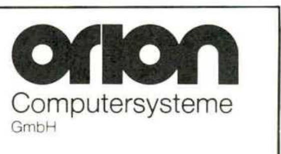
6457 Maintal



6500 Mainz - Hechtsheim



6520 Worms



6520 Worms · Friedrichstraße 22
Telefon 0 62 41 / 67 57 - 58

6581 Niederbrombach

INFODAT GmbH
ATARI Computersysteme
Protar Center
Beratung - Softwareentwicklung - Individuallösungen
Unter Hochkastell 3 Telefon:
6581 Niederbrombach (06787) 1425

6630 Saarlouis



6700 Ludwigshafen

MKV Computermarkt

Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

THEILLE
Computersysteme
Gilgenstraße 4 · 6720 Speyer
Telefon (0 62 32) 77 21 6

Etzkorn
Die Fachleute für Computer
Autorisierter ATARI - Vertragshändler
Hardware, Software & Zubehör für
• Apple • Acorn • Jonathan • MS/Dos •
6720 Speyer Auestraße 20
Telefon 06232 /32428 oder 32435
Mo.-Fr. 9 - 12 u. 15 - 18.30, Sa. 9 - 14 Uhr

6750 Kaiserslautern

INFODAT GmbH
ATARI Computersysteme
Protar Center
Beratung - Softwareentwicklung - Individuallösungen
Schubertstr. 16 6750 Kaiserslautern
Tel: (0631) 63597 Fax: (0631) 63589

6800 Mannheim



Computersysteme + Textsysteme

6800 Mannheim 24

Casterfeldstraße 74-76

☎ (06 21) 85 00 40 · Teletex 6 211 912

6900 Heidelberg

JACOM FAMILA-CENTER

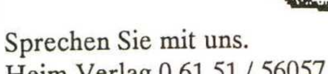
Hardware · Software
Schulung · Service

Hertzstraße 1 · 6900 Heidelberg 1
Telefon (0 62 21) 30 00 44-5

7000 Stuttgart



Bei uns werben bringt
GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

7030 Böblingen

Verkauf – Service – Software

Norbert Hlawinka
Sindelfinger Allee 1
7030 Böblingen
Tel. 0 70 31 / 22 60 15



COMPUTER CENTER

7047 Jettingen

Verkauf – Service – Software

Norbert Hlawinka
Heilbergstraße 3
Im Multi-Center
7047 Jettingen
Telefon (0 74 52) 7 76 15



COMPUTER SHOP

7100 Heilbronn

Computer-Welt

Seel's

Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 02

Hohmann & Co



Mönchseestraße 99
7100 Heilbronn
Telefon: 07131/60048

7150 Backnang



7312 Kirchheim/Teck

Comp & Phone

Computerfachhandel

Alleenstraße 66
7312 Kirchheim/Teck
Tel.: 07021/3949 - Fax: 07021/53933

7410 Reutlingen

MKV GMBH

Listplatz 2
7410 Reutlingen
Telefon 0 71 21 - 3 66 47

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis

HEIM + PC-COMPUTERMARKT

HARDWARE · SOFTWARE · LITERATUR

SCHEURER

ATARI COMMODORE CUMANA DATA-BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON

7475 Meßstetten 1 · Hauptstraße 10 · 0 74 31 / 6 12 80

7500 Karlsruhe

MKV GMBH

Kriegsstraße 77
7500 Karlsruhe
Telefon (07 21) 8 46 13

ERHARDT Am Ludwigsplatz

Am Ludwigsplatz · 7500 Karlsruhe 1 · Tel. (07 21) 16 08-0

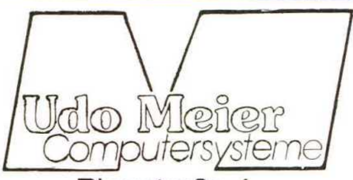
7600 Offenburg

FRANK LEONHARDT ELECTRONIC

Ihr Fachgeschäft für Microcomputer · Hifi · Funk

In der Jeuch 3
7600 Offenburg
Telefon 07 81 / 5 79 74

7700 Singen



Ringstraße 4
Telefon (0 77 31) 6 82 22

7730 VS-Schwenningen

BUS BRAUCH & SAUTER
COMPUTER TECHNIK

Villinger Straße 85
7730 VS-Schwenningen
Telefon 0 77 20 / 3 80 71-72

7750 Konstanz

neu: ATARI Desk Top Publishing-Center

computer - fachgeschäft

rösler

Rheingutstr. 1 • ☎ 0 75 31-2 18 32

7890 Waldshut-Tiengen



service gmbh

Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 0 77 51 / 30 94

7910 Neu - Ulm



7910 Neu-Ulm
Wegenerstr. 1
Tel. (07 31) 81 018 • Fax 86 833

7918 Illertissen

bictech gmbh

technische Informationsysteme
Computerladen

Marktplatz 13
7918 Illertissen
0 73 03 / 50 45

7930 Ehingen



7930 Ehingen
Thalstraße 12
Tel. (0 73 91) 80 28 • Fax 14 73

8000 München

Ihr Spezialist für:
dBMAN - Komplettlösungen
vortex Massenspeicher

Vortex, dBMAN sind eingetragene Warenzeichen zugunsten Dritter

A B A C München
Kellerstraße 11, 8000 München 80
Tel. 089/ 448 99 88



COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STRASSE 62L
EURO INDUSTRIE-PARK • 8000 MÜNCHEN 45
TELEFON 089 / 31 81 95-0 • TELETEX 89 83 41

City Studio
Rindermarkt 6, 8000 München 2
Tel. 089 / 2 60 98 01

schulz computer
Schillerstraße 22
8000 München 2
Telefon (0 89) 59 73 39

Beratung · Verkauf · Kundendienst

T.S. Service
Szemere Hard & Software
Schleißheimerstr.220
D-8000 München 40



Tel. 089/3089408 Fax 089/3085636 ATARI
Festplatten / Rechner / Zubehör / Service
Bürozeiten: Mo. bis Fr. 9.00-12.00 13.00-18.30 Samstag 9.00-14.00

8032 Gräfelfing



Pasinger Straße 94 8032 Gräfelfing
Telefon 089 - 8548823
Fax 089 - 8541764

8150 Holzkirchen



ATARI-SCHNEIDER-NEC
PANASONIC-PHILIPS
TOSHIBA PORTABLE-LUCKY GOLDSTAR

Hard- und Software in großer Auswahl

Service und Beratung sind bei uns Inklusive

8200 Rosenheim



COMPUTER + BÜROTECHNIK
Kufsteiner Str. 11, 8200 Rosenheim
Tel. (08031) 38 00 30
Fax (08031) 1 53 34

Autorisierter ATARI System-Fachhändler
Fischer & Bach
Computer GmbH
Münchner Straße 41 · 8200 Rosenheim
Tel.: 0 80 31 / 1 47 55 · Fax: 0 80 31 / 1 76 67
Hardware · Software
Schulung · Schnellreparatur

8390 Passau



8390 Passau
Kohlbruck 2a
Tel. (08 51) 52 007 • Fax 55 687

8400 Regensburg



im ELEKTROLAND REGENSBURG
Dr.-Gessler-Straße 8
Tel. (09 41) 9 50 85 · Fax 947961

8423 Abensberg

WITTICH
COMPUTER GMBH
Tulpenstr. 16 · 8423 Abensberg
☎ 0 94 43 / 4 53



8480 Weiden



8480 Weiden
Obere Bauscherstraße 21
Tel. (09 61) 46 105 • Fax 45 928

8500 Nürnberg



8500 Nürnberg
Färberstraße 20
Tel. (09 11) 20 80 84 • Fax 22 19 47

8520 Erlangen

Computerservice Decker
Meisenweg 29 - 8520 Erlangen
Telefon 0 91 31 / 4 20 76

Zimmermann elektroland

8520 Erlangen
Nürnberger Straße 88
Tel. (0 91 31) 3 45 68

8500 Nürnberg
Hauptmarkt 17
Tel. (09 11) 2 07 98





Erfolgreich werben

Sprechen Sie mit uns.
Heim-Verlag ☎ (0 61 51) 5 60 57 BUF

8520 Erlangen



8600 Bamberg





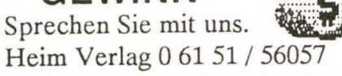
8700 Würzburg



Hardware · Software
Service · Schulung
computer center
am Dominikanerplatz
Ruf (09 31) 3 08 08 - 0

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057
BUF

8720 Schweinfurt



Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 65 21 54

8900 Augsburg



Unser Plus: Beratung u. Service
Schwalbenstr. 1 · 8900 Augsburg-Pfersee
Telefon (08 21) 52 85 33 oder 52 80 87
Computer Vertriebs- und Software GmbH



8906 Gersthofen



8940 Memmingen



8960 Kempten



ÖSTERREICH

A-1030 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m b H
A-1030 Wien
Landstraßer Hauptstraße 2
Hilton-Einkaufspassage

A-1040 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m b H
A-1040 Wien · Paniglgasse 18-20
Tel. (02 22) 5 05 78 08, 5 05 88 93

A-1180 Wien

Ihr kompetentes Atari-Fachgeschäft in Wien



Tel. (0222) 408 52 56
Fax: (0222) 408 99 78
A-1180 Wien - Schulgasse 63

SCHWEIZ



Computer Trend




Ihr Computer Spezialist
5000 Aarau, Bahnhofstrasse 86, Tel. 064/22 78 40
4102 Basel-Binningen, Kronenplatz, Tel. 061/47 88 64
5430 Wettingen, Zentralstrasse 93, Tel. 056/27 16 60
8400 Winterthur, St. Gallerstrasse 41, Tel. 052/27 96 96
8021 Zürich, Langstrasse 31, Tel. 01/241 73 73
Grösste Auswahl an Peripherie, Software, Literatur und Zubehör.

Kleinanzeigen

Biete Software

GFA-BASIC 3.5 EWS	238 DM
GFA-BASIC 3.0 EWS	168 DM
GFA-ASSEMBLER	128 DM
GFA-DRAFT Plus	328 DM
OMIKRON Basic-Compiler	169 DM
DRAW 3.0xx	119 DM
OMIKRON ISYGIM	75 DM

Public-Domain Disketten ab DM 2,50 lieferbar!
Katalog-Diskette gegen 3,- DM in Bfm.
**** 24 Stunden Schnellversand ! ****
TITAN-DATA, Elisabethstr. 36, 4040 Neuss 21
Tel. 0 21 07 / 75 95 Mo-Fr 10 – 17 Uhr

*** MASKEDIT ***
Entwerfen Sie Ihre Masken für ADITALK immer noch auf dem Weg EDITOR-TALKER-ADITALK? Dann ist MASKEDIT etwas für Sie, denn Sie können sich den Umweg über den Editor sparen. MASKEDIT erzeugt direkt TLK-Programme. Preis 69 DM. Demo 20 Reichel, Bühlstr. 8, 7507 Pfinztal 2, 0 72 40 / 75 69

*** TALKSHELL ***
Arbeiten Sie mit TALK und haben Probleme verschachtelte IF-, DO WHILE-, ROLL-Anweisungen zu überschauen? Dann könnte TALKSHELL für Sie eine ERleichterung bei der Fehlersuche sein. TALKSHELL markiert zusammengehörige Konstrukte, weist Sie auf Fehler hin, erlaubt ein komfortables Editieren, Compilieren... Preis 79 DM. Reichel 0 72 40 / 75 69

a) DB-System Adimens STI + Adiprog STI (Mark Williams C-Compiler) Netzwerk-Version für Atari ST günstig abzugeben wegen Fehlinvestition (gekauft 07/89, kaum genutzt) NP 3.500,- DM

b) Tape Streamer für 20 MB (Fa. Xebec) mit Schnittstelle DMA-SCSI kaum genutzt, gekauft 03/88, NP 2.000,-

Angebote bitte an Ralf Gerike, Reichsstraße 100, 1000 Berlin 19, Tel: 0 30 / 3 05 30 91

Original ST PASCAL+, GFA Ass. Verkaufspreis Tel. 0 93 23 / 30 19

FIBUSTAT 250,-, IMPORTMODUL 99,- • MEGAMAX MODULA 2 250,- • SUPERBASE PROFESSIONAL 299,- • TIMEWORKS DT-PUBLISHER 120,- • EASY DRAW 150,- Tägl. ab 18.00 Uhr 0 94 45 / 78 27

NEC P6/+TREIBER f. WORD+ KORREKTER SEITENUMBRUCH
Monitor = Drucker b. 1,5+1zlg.
BLOCK+PROP; ideal f. Ex-Arb; Dissert. * 02 03 / 55 77 29

LIGA.PRG U.A.ENGL.FUSSB.SYSTEM, SCHACH-USW.;NUR GEGEN V-SCHECK ÜBER 59 DM AN PETER BERNHARDT SANDKRUG 16, D-2300-KIEL-14

Ausschreibung-Angebot-Rechnung Datanorm/Baustoffverwaltung K-Zahlberechnung-Wärmebedarf Wärmeschutz-Dampfd./Tauwasser Dipl. Ing. V. Koch, Germersweg 21 3563 Dautphetal 3, Tel. 0 64 68 / 76 52

PD-SOFTWARE bis NR 333
Alle ST-Disks je DM 3,-
P+V 3,- DM / NN +5DM
N. TWARDOCH WNKENSTR. 67
4902 BAD SALZUFLEN

Softstation
2 PD's von ST-Comp.
Fortl. Nr. (1-2 usw) DM 5,-
Bel. Komb. DM 7,-
sonstige Serien DM 7,-
Disketten wahlw. blau, weiß, rot, grün, gelb od. orange
Verp. Porto plus NN DM 7,-
Tel. 0 71 95 / 5 37 07

ST-COMPUTER-PDs
SS 3DM/DS 4DM/P+V 5DM/NN+2DM
Katalogdisk 3DM bei: S.Heigert
Berndesallee 6, 6501 Heidesheim

MODELLBAHNSTEUERUNG MIT DEM ST mit Fahrplänen oder direkt im Gleisbild, Fahrstr., Anfahrverz., Ansteuerung im Interrupt... Info bei: Martin Birn, Knoblochstr. 51, 7100 Heilbronn

Kleinanzeigen

Biete Software

Public-Domain-Software aus der ST-Comp., 2000er, GFA u.a. je 3,40 DM inkl. 2S/2D-Disk Lieferung innerh. 24 Stunden! Liste auf Disk lieferbar Tel.: 0 40 / 7 12 54 97

Super PD Software für Atari ST Katalog DM 2,- in Briefmarken C. Schuegraf, Raibacher Tal 27 6114 Groß-Umstadt

---

*** HAUFENWEISE PD *** Ca. 400 PD-Disks vorhanden. ST-Computer, PD-Pool, Signum, Demos usw. Ab 2,50 DM je Disk (PD+Disk) Pfisterer, Hinterer Markt 1, 8411 Beratzhausen

---

HALLO MUSIKFANS, DEUTSCHE CHART SINGLES-AUSWERTUNG' 56-87 >10000 TITEL, ADI-DATEI, ASCII DM 40,-, GLAW W., 7700 SINGEN 16, TEL. 0 77 31 / 4 78 46 NACH 18.00 UHR

G-I-R Software Vertrieb Peter Grantz, Hauptstr. 49 D-2401 Ratekau/Lübeck

PD-Software für Atari ST,
- über 300 Disk.
  Angebot wird lfd. erweitert.
- Update Service
- 24-Std.-Auftragsdienst gewährleistet durch
Tel.: 0 45 04 / 41 15
BTX: 0 45 04 / 52 67

PD-Software Je Disk 4,50 Porto/Verp./NN 6,- SIGNUM-Pakete / PD-ABO. KURZ-INFO kostenlos bei: B. Jürgensen Hard & Soft Holmberg 4 / 2398 Harrislee

---

Anwender- u. Spielesoftware Laufend n. Programme a. Lager Riesenauswahl an Original Soft- u. Hardware Gratiskatalog anf. o. anrufen W. Wünsch Soft- u. Hardware, 0 72 31 / 76 65 95 Friedenstr. 212, 7530 Pforzheim

---

Wärmebedarf DIN 4701 • Wärmeschutz V • K-Zahl • Rohrnetz & Heizkörperauslegung • Demodisk DM 10,- Vorkasse • von J. Binder Behrensstr. 16, 5030 Hürth

---

PD-Software zu Traumpreisen! 4 komplette PD's (beliebig kombinierbar) inkl. Diskette für nur 5,50 DM Tel.0 27 21 / 24 32

ETIKETTEN DRUCKPROGRAMM für ST für alle Arten von Etiketten, frei initialisierbar, 128 Schriften, Kursiv, Fett - etc. DM 59; incl MwSt, Nachnahme +DM 5,00 Demodisk DM 10,- M. Reiter, Joh.-Gutenberg-Str.5 D-6256 Villmar

---

PUBLIC-DOMAIN-SOFTWARE Für PD-Freaks und Anwender. Bei uns möglich: Zusammenstellen nach KByte, Kopie auf unsere /Ihre, single/double, Marken-/NN-Disk, Tausch u.v.m. Große Auswahl! Gratiskatalog bei: Olaf Schwede, Röntgenweg 9/1, 7050 Waiblingen

---

Tempus Word (!) 500,- PC-Ditto 70,-, 1st Proportional 60,- Star Writer ST 60,- M&T: ST Pascal 30,- ST Computer 6/87 bis 4/90 80,- 04 31 / 55 25 68

BTX-Manager V3.0 für Postmodem + Interface, 250,- DM 0 61 55 / 6 36 90

Signum2 - DM 300,- +++ STAD 1.3 - DM 100,- +++ C-Entwicklungspaket Mark Williams DM 300,- PC-Ditto DM 300,- 0 70 31 / 80 21 81

Merkator ST FiBu - Preiswertes Werkzeug für Ihre Buchführung (DM 298,-, monochrom, GuV und Bilanz, Erscheinen: Juli). Demo anfordern! (DM 20,-, Verrechnungsscheck an **Verlag 1600 Marcus Daniel Cremer**, Pf 660, 4390 Gladbeck)

Biete Hardware

VERK. GEB. MAUS (50,-) Tel: 0 43 15 / 6 92 16

MEGA ST2, GS-Multisync-Monitor MEGAFILE 30, Software u. Lit. Preis VS, Tel. 02 01 / 66 67 02

Rest aus Sammelbestellung: **ST-RAM-ERWEITERUNG** um 2MB DM 400,- Tel.: 02 41 / 7 73 01

MEGA-ST-Tastaturen, Leergehäuse, Floppies, Netzteile, Mäuse neu u. original. Rauch 0 70 31 / 5 26 07

I/O-INTERFACE ITF-32 TTL-komp. 32 Ein- u. Ausgänge, erweiterbar f. Alarmanlage, Ind.-Steuerungen J. Sieber, Raffeltersteige 15 7100 Heilbronn, T. 0 71 31 / 16 11 51

Floppy-Station SF 314 DM 250,- SF 354 DM 60,- Tel: 0 82 38 / 39 87

Hayes-komp. MODEME v. ASTA
2400E(300,1200,2400 Baud) 278,-
2400E+(+BTX-NORM 1200/75) 398,-
2400 MNP 5 (4800 eff.) 498,-
NEU: 2400 MNP 5+ 598,-
BTX-Manager V. 3.02 248,-
Anschluß der Modeme am Postnetz der DBP Telekom ist strafbar! CSR, Breslauer Str. 19, 3575 Kirchhain, Tel.: 0 64 22 / 34 38

I/O-Interface-Technik inkl. Treibersoftware u. Anwendungsschaltungen für ATARI Info: K. Kellner, Adalbert-Stifter-Str. 12 3558 Frankenberg, Tel: 0 64 51 / 2 34 00

Atari ST 2MByte RAM-Erweiterung Leerplatine (industrielle Fertigung) 50DM **nur Vorkasse** (Verrechnungsscheck) F.Heyer Promenadenstr. 50, 5100 Aachen **Info: 02 41 / 3 52 47**

1040 STF + Uhr, SM124, SF314, Maus, Megafile 30 + Software: z.B. LaserC Pascal+, GFA-BASIC, Tempus, 1st Word, Flugsimulator II, 40 PD u.a. komplett VB DM 2500,- Weiß Th. Blütenstr. 16, 8000 Mü 40

65MB LACOM SCSI HD, ANSCHLUßF. **NEU 1133,- *** 1MB FLOPPY 180,- TOS 1.4 59,- 02 12 / 20 86 13

TOS 1.4, 2 ROMs 90,-, TOS 1.0+1.4, umschaltbar 120,-, ST Schaltplan 40,-, Calamus 1.09 500,- Tel. 0 69 / 5 07 48 10 18-20h

---

ATARI 520ST+ (1MB,ROM-TOS) + Mon. SM 124, 2xFloppy 720KB,mod. CSF-Gehäuse(Thermolüfter),Maus für VB 1500,- DM, Tel. 0 21 34 / 5 50 50

############ LA 4825 ############
Black-Box-Logikanalysator mit 48 Kanal bei 25 MHz ab 2470,90 DM incl. Für PC's, Atari, MC 68000. Disassembler nachrüstbar, bei: C.P.H. Tel. 0 74 23/60 66 Fax 60 67
############ LA 4825 ############

Speichererweiterung ST 260/520 auf 2,5 MB: 490,- DM / 560,- DM auf 4 MB: 980,- DM / 1120,- DM 07 21 / 37 66 22 excl. / incl. Einbau

---

Systemwechsel Mega ST 2 + TOS 1.4, SM 124, Protar 40 DC (nur 2 Monate alt), statt 4450,- jetzt nur noch 3800,- Tel. 0 71 21 / 33 82 23

SCSI-LW QUANTUM P80S 1000,- DM Rodime 100MB 18ms 3,5" 1100,- DM ST-Computer alle Ausg. 170,- DM Tel. 0 20 41 / 6 42 04 ab 19 Uhr

---

260 ST 1MB komplett mit SM 124 Star NL10 + 2 Laufwerke Wegen Systemwechsel verkauft Tel.0 28 42 / 4 18 26

MEGA ST 1 mit TOS 1.4 - 950 DM Dito m. 3MB -1450 • **50MB SCSI HD** m. Soft -1033 • 02 12 / 20 86 13

Scanner MICROTEK MSF-300 C inkl. Atari-DMA Interface und Software für DM 2000,- abzugeben. Daniela Ostermann Telefon: 02 41 / 2 96 37

MATRIX M110 inkl. Contr. DM 3600,- **HAWK COLIBRI SCANNER** 400 dpi 105mm inkl. OCR NP 1400,- VHB 850,- Tel. 06126-51732

ROM TOS 1.4 55,- Tel. 02 12/20 86 13

ATARI MEGA ST2 MEGAFILE 60 MIT TASTATUR BILDSCHIRM SM 124 MAUS NEUWERT DM 4700,- ZU DM 3500,- Chiffre Nr. 4011

Verschiedenes

ST-Computer 1/86-1/90 zu verk. Tel.: 0 24 43 / 38 83 • 150,- DM

Suchen für unsere Atariwerkstatt und unsere Entwicklungsabteilung Atari-Hardwareprofi, der bereits einige kleine Entwicklungen selbst erstellt hat. Einzelne Aufgabenbereiche sollten nach kurzer Einarbeitungszeit selbständig übernommen werden. Bewerbungsunterlagen erbeten unter Chiffre Nr. 4012

Vermittle jede gebrauchte Atari Hard- u. Originalsoftware, zu Höchstpreisen! Info bei R. Heinzelmann Egelsee 3, 7940 Altheim (Briefmarke DM 1,70 beifügen) 100 PD-Programme DD-Disk DM 4,- n.Wahl

---

Vielseitiger junger Club sucht Mitarbeiter zur Verstärkung des Teams im Raum Esslingen. Tel. 0 70 21 / 4 47 59 (Gerd)



INHALT



DIREKTE BLITTER-PROGRAMMIERUNG

ODER "BITTE EIN BLIT"

Andreas Hollmann

In diesem Artikel soll erklärt werden, wie man ohne (zeitraubenden) Umweg über das Betriebssystem durch direktes Setzen der Blitter-Register einen Bitblock-Transfer durchführen kann.

Zuerst werden die benötigten Register des Blitters mit Namen bezeichnet, dabei habe ich mich an die Bezeichnungen aus dem 'Atari ST Profibuch' gehalten:

Name	Typ	Adresse
src_xinc	W	$FF8A20
src_yinc	W	$FF8A22
src_addr	L	$FF8A24
endmask1	W	$FF8A28
endmask2	W	$FF8A2A
endmask3	W	$FF8A2C
dst_xinc	W	$FF8A2E
dst_yinc	W	$FF8A30
dst_addr	L	$FF8A32
x_count	W	$FF8A36
y_count	W	$FF8A38
hop	B	$FF8A3A
op	B	$FF8A3B
line_num	B	$FF8A3C
skew	B	$FF8A3D

Tabelle 1: Die benötigten Register des Blitters

Das Halftone-RAM zum Laden der Füllmusterdaten besteht aus 16 Datenwörtern, die an den Adressen $FF8A00 bis $FF8A1E liegen. Die ersten drei Blitter-Register beschreiben den Aufbau des Quell-Rasters:

src_xinc muß den Abstand zwischen zwei Datenwörtern einer Zeile enthalten; es wird dadurch indirekt die Anzahl der Bitplanes angegeben. In hoher Auflösung steht hier 2, in mittlerer 4 und in niedriger 8. *src_yinc* enthält den Adreßabstand in Bytes zwischen dem letzten Wort einer Zeile und dem ersten der nächsten. *src_addr* enthält die 4Basisadresse des Quellbildschirms. Die drei '*endmask*'-Register werden intern für den Transfer benötigt:

endmask1 enthält ein Bit-Muster, aus dem der Blitter erkennt, welche Bits im ersten Zieldatenwort geändert werden müssen. Ist die x-Koordinate des Ziel-Bit-Blocks ein durch 16 teilbarer Wert, müssen nur ganze Worte verarbeitet werden, in *endmask1* steht dann der Wert

%1111111111111111;

es müssen also alle 16 Bits des Zielwortes geändert werden. Ist der Bit-Block z.B. um 1 Pixel weiter nach rechts geschoben, darf das 1. Bit des 1. Zieldatenwortes nicht bearbeitet werden; in *endmask1* steht dann der Wert

%0111111111111111.

endmask2 und *endmask3* haben die gleiche Funktion, nur bestimmt *endmask3* die Bits, die im letzten Zieldatenwort einer Zeile bearbeitet werden sollen, und *endmask2* die zu bearbeitenden Bits in allen dazwischenliegenden Datenwörtern.

Die drei Register *dst_xinc*, *dst_yinc* und *dst_addr* beschreiben den Aufbau des Zielrasters (Bedeutung wie bei den *src_...*-Registern oben). Die Register *x_count* und *y_count* beschreiben den zwischen Quell- und Zielraster zu kopierenden Bit-Block:

x_count enthält die Anzahl der Datenwörter einer Zeile. Ist die Breite des Bit-Blocks nicht ohne Rest durch 16 teilbar, wird der erhaltene Wert nach oben gerundet. Welche Bits dann tatsächlich bearbeitet werden, wird wie oben beschrieben durch die drei *endmask*-Register festgelegt. *y_count* enthält die Anzahl der Zeilen des Bit-Blocks.

Das *hop* (=Halftone-Operation)-Register bestimmt die Verknüpfung zwischen Quelldaten und dem Blitter-internen Halftone-RAM. Steht hier eine 0, werden alle Bits des Ziel-Bit-Blocks auf 1 gesetzt (was soll das?...).

Bei dem Wert 1 werden nur Daten aus dem Halftone-RAM geschrieben; das muß mal näher erklärt werden: Dieser Modus ist da, um mit dem Blitter Flächen zu füllen. Dazu wird das Füllmuster in das Halftone-RAM geladen; ein Füllmuster besteht aus 16*16 Pixeln, es paßt also (welch ein Zufall!) genau in das Halftone-RAM hinein. Der Blitter kümmert sich in diesem Modus gar nicht um den Quellbildschirm, sondern kopiert das Füllmuster aus dem Halftone-RAM in den Ziel-Bit-Block.

Bei einem Wert von 2 kümmert sich der Blitter wiederum nicht um das Halftone-RAM, sondern kopiert nur den Quell-Bit-Block in den Zielbildschirm - das ist der Modus, den man für einen 'normalen' Bit-Block-Transfer braucht! Der Überflüssigkeit halber gibt's noch den HOP-Modus 3: Hier werden Quell- und Halftone-Daten logisch AND-verknüpft.

Das 'op'-Register bestimmt die Verknüpfung von Quell- und Zieldaten: Die hier zulässigen Werte von 0-15 haben folgende Bedeutung:

Modus	Verknüpfung	Bedeutung
0	0	mit 0-Bits füllen (weiß)
1	src AND dst	logisch AND
2	scr AND (NOT dst)	
3	src	replace, Standardmodus
4	(NOT src) AND dst	
5	dst	Es passiert nichts! (toll, nä ?)
6	src XOR dst	
7	src OR dst	transparent
8	NOT (src OR dst)	invertiert
9	NOT (src XOR dst)	
10	NOT dst	
11	src OR (NOT dst)	
12	NOT src	
13	(NOT src) OR dst	
14	NOT (src AND dst)	
15	1	mit 1-Bits füllen (schwarz)

Tabelle 2: Die zulässigen Werte des 'op'-Registers

Die Ergebnisse der meisten Verknüpfungsmodi lassen sich schlecht mit Worten beschreiben, einfach ausprobieren und sich überraschen lassen!

Das *line_num*-Register ist ein interner Zähler für den Blitter und enthält das gerade benutzte Halftone-Wort. Hierfür werden die unteren 4 Bits des 1 Byte breiten Registers benutzt. Das oberste Bit (7) 'busy' startet den Bit-Block-Transfer und sollte erst dann gesetzt werden, wenn alle anderen Register gefüllt sind (sonst denkt sich der Blitter selbst was aus, und das hat meistens einen Bombenerfolg).

Das Bit Nr. 6 alias *hog* (nicht mit *hop* zu verwechseln!) regelt den DMA-Zugriff von Blitter und 68000er-CPU: Ist es gesetzt, hat der Blitter eine höhere Priorität als die CPU; diese wird solange angehalten, bis der Blitter den Bit-Block-Transfer beendet hat. Ist das Bit gelöscht, wechseln sich Blitter und CPU alle 64 Zyklen ab.

Das Bit Nr.5 alias *smudge* kann für Schmiereffekte benutzt werden: Ist es gesetzt, bestimmen nicht die 4 Bits *line_num* das aktuelle Halftone-Datenwort, sondern die 4 niedrigsten Bits der verschobenen Quelldaten.

Zum Schluß noch das *skew*-Register: Wird ein Bit-Block z.B. um 1 Pixel nach rechts verschoben, muß hier eine 1 stehen. Der maximale Wert ist 16 (dann befindet man sich auf der nächsten durch 16 teilbaren x-Koordinate).

Nun geht's los!

Die erste Disziplin ist horizontales Scrolling. Ohne Blitter dauert das horizontale pixelweise Scrollen eines Bilds mit dem BITBLT-Befehl (GFA- und Omikron.BASIC) 116 s (während des Scrollens kann gefrühstückt werden), mit eingeschaltetem Blitter ca. 14 s! Aha, denkt man erfreut, der Blitter bringt's ja wirklich.

Da bei Benutzung des BITBLT-Befehls aber das Betriebssystem seine Finger drin hat, müßte durch direkte Blitter-Programmierung noch eine höhere Geschwindigkeit erreichbar sein.

Die Assembler-Routine 'BLT_RT.Q' bringt dem Blitter das horizontale Schieben nach rechts durch direktes Setzen seiner Register bei (läuft in hoher Auflösung). Als einziger Übergabeparameter wird die Adresse des physikalischen Bildschirms erwartet, welcher über den Stack übergeben wird (man könnte ihn natürlich auch innerhalb der Routine ermitteln). Nachdem allen Blitter-Registern die Namen zugeordnet wurden, wird die CPU in den Supervisor-Modus geschaltet, da nur dann auf die Hardware-Register zugegriffen werden kann (Label: *go_super*; beim Aufruf aus Omikron.BASIC kann diese Routine weggelassen werden, da sich Omikron.-BASIC bereits im Supervisor-Modus befindet). Es folgt die eigentliche Hauptroutine: die Blitter-Register werden mit den entsprechenden Daten geladen und der Blitter mit gesetztem HOG-Bit gestartet. Da die Timer-Abfrage falsche Werte liefert, muß von Hand gestoppt werden. Das Ergebnis: 10.6 s (Junghans-Quartz-gestoppt) - also deutlich schneller als beim Aufruf über das Betriebssystem.

Und nun der Geschwindigkeitsvergleich mit der CPU: Die Assembler-Routine 'CPU_RT.Q' schiebt den Bildschirm horizontal pixelweise nach rechts. [Die Routine wurde mit dem Omikron.-Assembler geschrieben. Die beiden Befehle 'REPT n' und 'ENDR' sind Assembler-Direktiven; der Befehl zwischen diesen Befehlen wird n-mal (hier 40mal) wiederholt. (Wer mit einem Assembler arbeitet, der diese oder ähnliche Direktiven nicht kennt, kopiert einfach 40mal *roxr.w (a6)+* untereinander.)] Mit dieser Routine dauert das pixelweise Schieben 16.8 s, also gar nicht mal so viel langsamer, wie man erwartet hätte.

Hatte man beim ersten Vergleich mit ein- und ausgeschaltetem Blitter einen Geschwindigkeitsgewinn um den Faktor 8 gemessen, schrumpft dieser Vorsprung bei genauerer Untersuchung auf den Wert 1.6 zusammen. Die große Differenz im ersten Fall liegt wohl weniger an der Schnelligkeit des Blitters, sondern an der langsamen BITBLT-Software-Emulation durch das Betriebssystem.

Beim vertikalen Scrollen sieht es noch ungünstiger aus: einen ganzen Bildschirm pixelweise nach oben schieben dauert mit der Assembler-Routine 'CPU_UP.Q' 7.9 s, mit direkt programmiertem Blitter ('BLT_UP.Q') 6.6 s. Geschwindigkeitssteigerung: zirka Faktor 1.2.

Beim wortweisen Zugriff auf den Speicher, wie es beim vertikalen Scrollen der Fall ist, ist der Blitter nur wenig schneller als die CPU. Sobald aber Schiebebefehle notwendig sind, gewinnt der Blitter gegenüber der CPU an Boden. Das kommt deutlicher zur Geltung, wenn man beim horizontalen Scrollen nicht in 1-, sondern in 2- oder 4-Pixel-Schritten arbeitet; dann verdoppelt/vervierfacht sich die Geschwindigkeit annähernd, d.h. die Schiebebefehle innerhalb des Blitters werden extrem schnell abgearbeitet.

Sicherlich ist der Bitblit-Algorithmus flexibler als spezielle Scroll-Routinen, aber bei einem Rechner ohne Hardwarescrolling hätte man zur komfortablen Programmierung in einer Hochsprache (wer hat denn immer Lust auf Maschinensprache...) zusätzlich schnellere Scroll-Befehle implementieren sollen. Nun kann man natürlich zur Verteidigung des Blitters noch die Vielfalt der Verknüpfungsmöglichkeiten anführen; ob die dann vielleicht erheblich schneller sind als mit dem 68000er, werde ich demnächst mal testen. Wenn ich den Grundgedanken, den Atari mit dem Blitter verfolgt, aber richtig verstanden habe, ist er u.a. als Ausgleich für die Rechenleistung schluckende GEM-Benutzeroberfläche gedacht und soll hier beispielsweise den Hintergrund von Dialogboxen oder Fenstern restaurieren. Bei diesen Funktionen werden aber die vielfältigen Verknüpfungsmöglichkeiten gar nicht benötigt. Man gewinnt aus den enormen Geschwindigkeitsunterschieden beim Vergleich der BITBLT-Routine mit ein- und ausgeschaltetem Blitter eher den Eindruck, daß hier langsame Software mit schneller Hardware ausgeglichen wird. Das würde auch die Beschleunigung durch das 'Software-Blitter-Accessory' 'Turbo-ST' erklären.

Was mich bei der Programmierung des Blitters etwas irritierte, war das sogenannte HOG-Bit (Bit 6 an Adresse $FF8A3C). Dieses Bit regelt den DMA-Zugriff von Blitter und CPU; bei gelöschtem Bit kommt jeder für 64 Zyklen an den Bus, bei gesetztem Bit hat

der Blitter Vorrang. Bei meinen Versuchen kam es bei schnell aufeinanderfolgenden Bit-Blockverschiebungen (wie es beim Scrollen der Fall ist) und nicht gesetztem HOG-Bit (move.b #%1000000, $FF8A3C) zum absoluten Chaos: der Blitter hat wie wild im Bildschirm und wahrscheinlich auch im restlichen Speicher herumgeschrieben, denn der nächste Absturz war hiermit wieder fällig. Mit gesetztem HOG-Bit (move.b #%11000000, $FF8A3C) war alles in Ordnung.

So, das war's erst mal zum Blitter. Ich hoffe, daß die Listings ihrem Zweck, als Anregung für eigene Routinen zu dienen, gerecht werden; sie laufen übrigens alle in hoher Auflösung. Die Einbindung in Hochsprachen sollte wohl keine Probleme bereiten. Das Listing 'BASIC.LST' zeigt z.B. die Einbindung in Omikron.- und GFA-BASIC.

```
 1:  ' ************************************************
 2:  ' *     BLT_DEMO.GFA - Demonstration für direkte
          Blitter-Programmierung                    *
 3:  ' ************************************************
 4:  RESERVE 8000                 ! 8 kB sind genug
 5:  IF MALLOC(-1)>=32000 THEN ! genug Platz für Bild?
 6:    p_bild%=MALLOC(32000)     ! aber sicher !
 7:  ELSE
 8:    PRINT AT(1,1);"Nicht genug RAM frei !"  ! nicht
 9:    RESERVE
10:    END
11:  ENDIF
12:  ' in die folgenden INLINE-Blöcke werden die
       Assembler-Codes geladen:
13:  INLINE blt_up%,176
14:  INLINE blt_rt%,172
15:  INLINE cpu_up%,52
16:  INLINE cpu_rt%,108
17:  BLOAD "a:\bild.scr",p_bild%          ! Bild laden
18:  '
19:  PRINT AT(18,1);"Demo-Programm für direkte
                       Blitter-Programmierung"
20:  PRINT AT(30,2);"von Andreas Hollmann"
21:  ~INP(2)
22:  show_message("Vertikales Scrolling mit dem
                   Blitter")
23:  BMOVE p_bild%,XBIOS(2),32000            ! Bild aus
                           Puffer in den Bildschirm
24:  ~C:blt_up%(L:XBIOS(2))           ! Assembler-Routine
25:  show_message("Vertikales Scrolling mit der CPU")
26:  BMOVE p_bild%,XBIOS(2),32000
27:  ~C:cpu_up%(L:XBIOS(2))
28:  show_message("Horizontales Scrolling mit dem
                   Blitter")
29:  BMOVE p_bild%,XBIOS(2),32000
30:  ~C:blt_rt%(L:XBIOS(2))
31:  show_message("Horizontales Scrolling mit der
                   CPU")
32:  BMOVE p_bild%,XBIOS(2),32000
33:  ~C:cpu_rt%(L:XBIOS(2)) ! Assembler-Routine
34:  '
35:  ~MFREE(p_bild%)          ! Bild-Puffer an GEMDOS
36:  RESERVE                  ! alles wie vorher
37:  END
38:  '
39:  PROCEDURE show_message(text$)
40:    CLS
41:    PRINT AT(1,1);text$;"     (weiter mit Taste...)"
42:    ~INP(2)
43:    CLS
44:  RETURN
```

Listing 1: Eine Demonstration des direkten Blittens

```
1:  ' GFA-BASIC
2:  INLINE blitter%,256
3:  BLOAD "BILD.PIC",XBIOS(2)       ! Bild laden
4:  ~C:blitter%(L:XBIOS(2))         ! Assembler-Routine
5:  END

1:  ' Omikron.BASIC
2:  MEMORY_BLOCK 00,256,blitter%L
3:  BLOAD "BLITTER.INL",blitter%L
4:  XBIOS(adr%L,2)                  ! Bildschirm-Adr.
5:  BLOAD "BILD.PIC",adr%L          ! Bild laden
6:  CALL blitter%L(L adr%L)         ! Assembler-Routine
7:  END
```

Listing 2: So werden die Inlines in GFA- und Omikron.BASIC eingebunden

```
 1:  ;************************************************
 2:  ;* BLT_RT - Demonstrations-Programm zum
        horizontalem Scrollen mit dem Blitter      *
 3:  ;************************************************
 4:  ;
 5:  src_xinc:    EQU $FF8A20    ;zuerst werden alle be-
 6:  src_yinc:    EQU $FF8A22    ;nötigten Blitter-Re-
 7:  src_addr:    EQU $FF8A24    ;gister mit Namen be-
 8:  endmask1:    EQU $FF8A28    ;zeichnet (sonst findet
 9:  endmask2:    EQU $FF8A2A    ;da ja keiner mehr
10:  endmask3:    EQU $FF8A2C    ;durch):
11:  dst_xinc:    EQU $FF8A2E
12:  dst_yinc:    EQU $FF8A30
13:  dst_addr:    EQU $FF8A32
14:  x_count:     EQU $FF8A36
15:  y_count:     EQU $FF8A38
16:  hop:         EQU $FF8A3A
17:  op:          EQU $FF8A3B
18:  line_num:    EQU $FF8A3C
19:  skew:        EQU $FF8A3D
20:  ;-----------------------------------------------
21:               bsr      go_super    ;ggf. in Super-
                                        visormodus
22:  ;-----------------------------------------------
23:               movea.l  4(sp),a6    ;Bildschirm-Adr.
                                        vom Stack holen
24:               move.w   #639,d0     ;640x 1 Pixel
                                        nach rechts
25:  rt_loop:     move.w   #2,src_xinc ;Quelldatenwrte
                                        2 Bytes Abstand
26:               move.w   #2,src_yinc ;Zl-Ende und
                                        -Anfang 2 Byte
                                        Abstand
27:               move.l   a6,src_addr ;Adresse Quell-
                                         Bildschirm
28:               move.w   #$7FFF,endmask1 ;erstes Da-
                                        tenwrt nur 0-14
                                        bearbeiten
29:               move.w   #$FFFF,endmask2 ;mittlere
                                        Datenworte kom-
                                        plett bearbeit.
30:               move.w   #$FFFF,endmask3 ;letztes
                                        Datenwort komp.
                                        bearbeiten
31:               move.w   #2,dst_xinc ;Ziel-Datenwrte
                                        2 Bytes Abstand
32:               move.w   #2,dst_yinc ;Zl-Ende & -An-
                                        fang 2 Byte
                                        Abstand
33:               move.l   a6,dst_addr ;Adresse Ziel-
                                        Bildschirm
34:               move.w   #40,x_count ;Bitblock 40
                                        Worte pro Zeile
35:               move.w   #400,y_count ;Bitblock 400
                                        Zeilen
36:               move.b   #2,hop      ;nur Source-Da-
                                        ten kopieren
37:               move.b   #3,op       ;Modus 3=Replace
38:               move.b   #1,skew     ;1 Bit rechts
                                        verschieben
39:               move.b   #192,line_num ;Blitter mit
                                        gesetztem HOG-
                                        Bit starten
40:               dbra     d0,rt_loop
41:               rts                  ;zurück zum
                                        Hauptprogramm
42:  ;-----------------------------------------------
43:  go_super:    pea      1           ;CPU-Status
44:               move.w   #32,-(sp)   ;Super
```

→

```
45:              trap     #1          ;GEMDOS-Aufruf
46:              addq.l   #6,sp       ;Stack korrig.
47:              cmpi.l   #-1,d0      ;d0=-1 ?
48:              beq.s    super_rts   ;JA: CPU bereits
                                       im Super
49:              pea      0           ;NEIN: CPU in
                                       Super schalten
50:              move.w   #32,-(sp)   ;Super
51:              trap     #1          ;GEMDOS-Aufruf
52:              addq.l   #6,sp       ;Stack korrig.
53: super_rts:   rts
54: ;------------------------------------------------
55:              END
```

Listing 3: Horizontales Scrolling mit dem Blitter

```
 1: ;************************************************
 2: ;*  BLT_UP - Demonstrations-Programm zum
       vertikalem Scrollen mit dem Blitter          *
 3: ;************************************************
 4: ;
 5: src_xinc:   EQU $FF8A20   ;zuerst werden alle be-
 6: src_yinc:   EQU $FF8A22   ;nötigten Blitter-
 7: src_addr:   EQU $FF8A24   ;Register mit Namen be-
 8: endmask1:   EQU $FF8A28   ;zeichnet (sonst
 9: endmask2:   EQU $FF8A2A   ;findet da ja keiner
10: endmask3:   EQU $FF8A2C   ;mehr durch):
11: dst_xinc:   EQU $FF8A2E
12: dst_yinc:   EQU $FF8A30
13: dst_addr:   EQU $FF8A32
14: x_count:    EQU $FF8A36
15: y_count:    EQU $FF8A38
16: hop:        EQU $FF8A3A
17: op:         EQU $FF8A3B
18: line_num:   EQU $FF8A3C
19: skew:       EQU $FF8A3D
20: ;------------------------------------------------
21:             bsr       go_super    ;ggf. in den
                                       Supervisormodus
22: ;------------------------------------------------
23:             movea.l   4(sp),a6    ;Bildschirm-Adr.
                                       vom Stack holen
24:             lea       80(a6),a5   ;Quelladresse
                                       berechnen
25:             move.w    #399,d0     ;400x um 1 Pixel
                                       nach oben
26: up_loop:    move.w    #2,src_xinc ;Quelldatenwrte
                                       2 Bytes Abstand
27:             move.w    #2,src_yinc ;Zl-Ende und
                                       -Anfang 2 Byte
                                       Abstand
28:             move.l    a5,src_addr ;Adresse des
                                       Quell-Screens
29:             move.w    #$FFFF,endmask1 ;erstes
                                       Datenwort kom-
                                       plett bearbeit.
30:             move.w    #$FFFF,endmask2 ;mittlere
                                       Datenworte kom-
                                       plett bearbeit.
31:             move.w    #$FFFF,endmask3 ;letztes
                                       Datenwort kom-
                                       plett bearbeit.
32:             move.w    #2,dst_xinc ;Zieldatenworte
                                       2 Bytes Abstand
33:             move.w    #2,dst_yinc ;Zl-Ende & -An-
                                       fang haben 2
                                       Byte Abstand
34:             move.l    a6,dst_addr ;Adresse des
                                       Ziel-Screens
35:             move.w    #40,x_count ;Bitblock 40
                                       Worte/Zeile
36:             move.w    #400,y_count ;Bitblock hat
                                       400 Zeilen
37:             move.b    #2,hop      ;nur Source-Da-
                                       ten kopieren
38:             move.b    #3,op       ;Modus 3=Replace
39:             move.b    #0,skew     ;keine Bits
                                       verschieben
40:             move.b    #192,line_num ;Blitter mit
                                       gesetztem HOG-
                                       Bit starten
41:             dbra      d0,up_loop
42:             rts                   ;zurück zum
                                       Hauptprogramm
43: ;------------------------------------------------
```

```
44: go_super:   pea       1           ;CPU-Status ab-
                                       fragen
45:             move.w    #32,-(sp)   ;Super
46:             trap      #1          ;GEMDOS-Aufruf
47:             addq.l    #6,sp       ;Stack korrig.
48:             cmpi.l    #-1,d0      ;d0=-1 ?
49:             beq.s     super_rts   ;JA: CPU ist
                                       schon im Super-
                                       visor-Modus
50:             pea       0           ;NEIN: CPU in
                                       Super schalten
51:             move.w    #32,-(sp)   ;Super
52:             trap      #1          ;GEMDOS-Aufruf
53:             addq.l    #6,sp       ;Stack korrig.
54: super_rts:  rts
55: ;------------------------------------------------
56:             END
```

Listing 4: Vertikales Scrolling mit dem Blitter

```
 1: ;************************************************
 2: ;*   CPU_RT - Demonstrations-Programm zum
        horizontalen Scrollen mit der CPU           *
 3: ;************************************************
 4:              movea.l   4(sp),a5    ;Bildschirm-
                                     -Adresse vom Stack
 5:              move.w    #639,d0     ;640mal um 1 Px
                                        nach rechts
 6: page_loop:   movea.l   a5,a6       ;Adresse restaur
 7:              move.w    #399,d1     ;400 Zeilen 1 Px
                                        nach rechts
 8: col_loop:    move      #0,ccr      ;X-Flag löschen
 9:              REPT 40               ;'roxr.w (a6)+
                                        40x wiederholen
10:              roxr.w    (a6)+       ;1 Wort um 1 Bit
                                        nach rechts
11:              ENDR
12:              dbra      d1,col_loop ;nächste Spalte
13:              dbra      d0,page_loop
14:              rts
15: ;------------------------------------------------
16:              END
```

Listing 5: Horizontales Scrolling mit der CPU

```
 1: ;************************************************
 2: ;*    CPU_UP - Demonstrations-Programm zum
         vertikalen Scrollen mit der CPU            *
 3: ;************************************************
 4:              movea.l   4(sp),a5    ;Bildschirm-Adr.
                                        vom Stack holen
 5:              move.w    #399,d0     ;400 mal 1 Pixel
                                        nach oben
 6: page_loop:   movea.l   a5,a6       ;Adresse restau-
                                        rieren
 7:              move.w    #399,d1     ;400 Zeilen 1
                                        Pixel nach oben
 8: line_loop:   movem.l   80(a6),d2-a3  ;1. Hälfte
                                      einer Zeile laden
 9:              movem.l   d2-a3,(a6)  ;1. Hälfte einer
                                        Zeile schreiben
10:              lea       40(a6),a6   ;Adresse 1/2
                                        Zeile weiter
11:              movem.l   80(a6),d2-a3 ;2. Hälfte ei-
                                        ner Zeile laden
12:              movem.l   d2-a3,(a6)  ;2. Hälfte einer
                                        Zeile schreiben
13:              lea       40(a6),a6   ;Adresse 1/2
                                        Zeile weiter
14:              dbra      d1,line_loop ;nächste Zeile
15:              dbra      d0,page_loop
16:              rts
17: ;------------------------------------------------
18:              END
```

Listing 6: Vertikales Scrolling mit der CPU

CLIPBOARD

Eric Böhnisch

WIR WOLLEN DIESEN TEXT MIT EINER SCHILDERUNG DER PROBLEME DES HERRN S. AUS K. BEGINNEN:
ER, SELBST COMPUTER-KENNER UND LEHRER AN EINEM GYMNASIUM, ENTDECKTE VOR EINIGER ZEIT AUF EINER MESSE EINEN COMPUTER, IN DEN ER SICH FAST VERLIEBTE. IHN FASZINIERTEN DER GRAUE, FLIMMERFREIE BILDSCHIRM, DIE GRAFISCHE BENUTZEROBERFLÄCHE UND DAS APFEL-ZEICHEN IN DER LINKEN OBEREN ECKE. DA ER ABER NICHT ALLZUVIEL GELD HATTE, ENTSCHIED ER SICH, SICH EINEN (IN MANCHEN PUNKTEN) VERGLEICHBAREN, ABER BILLIGEREN COMPUTER ZU KAUFEN. ER BESORGTE SICH ALSO DIESEN PLASTIKKASTEN UND WAR VOLLKOMMEN GLÜCKLICH, ALS DESSEN FUJIYAMA-ZEICHEN ANSTELLE DES APFELS IM MENÜBALKEN AUFLEUCHTETE. HERR S. AUS K. WAR GLÜCKLICH.

Dann jedoch er erinnerte sich an eine herrliche Sache an diesem großen Vorbild, an die tolle Möglichkeit des Datenaustausches zwischen zwei Programmen und an das schöne Klemmbrett-Bildchen - vergebens suchte er. Nichts! Das riß ihn aus seiner Traumwelt wieder zurück auf den Boden der Computer-Wirklichkeit.

Doch halt! Hat da nicht irgendjemand etwas vom den Scrap-Funktionen gesagt? Wie? 1st_Wordplus unterstützt eine Art Clipboard ab der Version 3.15? Bahnt sich da etwa eine Rettung für unseren Herrn S. aus K. an?

Das Klemmbrett

Das eigentliche Klemmbrett ist ein Ordner auf der Startdiskette oder der Startpartition der Festplatte. In ihm können GEM-Applikationen ihre Daten in einem standardisierten Datenformat ablegen (wobei man sich um den "Standard" sicherlich noch streiten kann und wird), und andere Programme können sie wieder herausholen und weiterverwenden. Somit können verschiedenste Applikationen Daten untereinander austauschen.

Woher wissen die Applikationen, wo dieser Ordner zu finden ist? Ganz einfach. Es gibt extra dafür einen GEM-Speicher, der mit den Funktionen *scrp_read()* und *scrp_write()* gelesen und beschrieben werden kann. In ihm steht normalerweise der Zugriffspfad auf den Ordner. Also, wenn man den Ordner CLIPBRD auf der Partition C: als Klemmbrett benutzen will, muß in diesem Speicher die Zeichenkette "C:\CLIPBRD" mit einem abschließenden Null-Byte stehen. Leider habe ich aus keiner der mir zur Verfügung stehenden Quellen erfahren können, wie groß der Puffer ist. Er reicht aber normalerweise für jeden verwinkelten Zugriffspfad (ich persönlich tippe auf 256 Bytes).

Will nun eine Applikation Daten aus dem Klemmbrett holen, liest sie diesen Speicher mit *scrp_read()* und schaut dann in dem beschriebenen Ordner nach.

Grundlegendes

Wenn eine Applikation gestartet wird, die das Klemmbrett unterstützt, sollte sie als erstes einmal nachschauen, ob sich etwas im Klemmbrett-Puffer befindet. Wenn nicht, wird ein Standardpfad von der Applikation eingerichtet und dann überprüft, ob dieser Ordner schon vorhanden ist. Wenn dies auch nicht der Fall ist, wird er neu angelegt.

Je nachdem, ob schon eine für die Applikation lesbare Datei im Klemmbrett liegt, sollten dann die entsprechenden Menüeinträge für "Aus Klemmbrett holen..." und "Ins Klemmbrett legen..." anwählbar (enabled) oder gesperrt (disabled) werden, damit keine Fehlbedienung möglich ist.

Im Klemmbrett an sich dürfen sich nur Dateien mit dem Namen *SCRAP.** befinden, wobei anstelle des Sterns eine kennzeichnende Dateinamenserweiterung gesetzt wird.

Die Standardformate und ihre Extensionen sind folgende:

SCRAP.TXT
ein ASCII-Text

SCRAP.1WP
ein Dokument im 1st_Wordplus-Format

SCRAP.IMG
ein Bild im GEM-Image-Format

SCRAP.GEM
eine GEM-Metadatei

SCRAP.CSV
ein ASCII-Text, der durch Kommas getrennte Zahlen enthält

SCRAP.DIF
eine Tabellenkalkulationsdatei

Man sollte möglichst nur diese Formate verwenden, denn wenn man irgendein programmspezifisches Format nimmt, kann man fast sicher sein, daß keine andere Applikation diese Daten lesen kann und das war ja eigentlich der Sinn der Sache.

Eine weitere Konvention ist es, daß, wenn eine Applikation ins Klemmbrett schreibt, sie vorher alle anderen Dateien dort zu löschen hat, da sonst die Zielapplikation ja gar nicht wüßte, welche *SCRAP.**-Datei sie zu laden hat. Sie würde annehmen, alle Dateien stammten von der Quellapplikation und sie würde das Format laden, das von ihr am besten zu verarbeiten ist. Aber genau diese Datei kann ja noch von einem vorhergehenden Klemmbrettzugriff eines dritten Programms stammen. Das führt zu nichts als Ärger beim Benutzer.

Initialisieren des Klemmbretts

Jede klemmbrettunterstützende Applikation sollte in der Initialisierungsphase auch das Klemmbrett initialisieren können, falls dies nötig sein sollte.

Dabei überprüft sie als erstes, ob sich schon etwas im GEM-Speicher befindet. Wenn nicht, überprüft man die Environment-Variable *CLIPBRD* (mit der Funktion *getenv*). Ist sie vorhanden, setzt



man den GEM-Puffer mit Hilfe von *scrp_write()* auf den Wert dieser Variablen. Sollte CLIPBRD nicht existieren, schreibt man einen vernünftig erscheinenden Wert hinein (üblicherweise "C:\CLIPBRD" bei Systemen mit Festplatte und "A:\CLIPBRD" bei Systemen ohne). Als letztes schaut man, ob dieser Ordner schon existiert. Wenn nicht, wird er angelegt [mit *mkdir()*]. Somit kann man sicher sein, daß, will man auf das Klemmbrett zugreifen, eins existiert.

Lesen aus dem Klemmbrett

Wenn eine Applikation aus dem Klemmbrett Daten lesen möchte, muß sie sich erst einmal selbst im klaren sein, welches Dateiformat sie akzeptieren kann; z.B. werden die meisten Textprogramme wohl ASCII-Daten verarbeiten können.

Dann holt sie sich mit Hilfe der Funktion *scrp_read()* den Klemmbrettpfad aus dem internen Puffer des GEM. An diesen hängt sie nun den Namen "\SCRAP." und dann die gewünschte Extension, für einen ASCII-Text beispielsweise "TXT", an. Besser geht dies jedoch mit den Bibliotheksfunktionen *strmfp()* und *strfmc()*, wie sie z.B. der Lattice-C-Compiler V3.04 bietet.

Und mit diesem Namen kann dann die Datei, wenn sie vorhanden ist, geladen und verarbeitet werden. Sollte die Datei nicht existieren, so sollte nicht die Meldung "Datei nicht gefunden", sondern besser "Keine für diese Applikation lesbaren Daten im Klemmbrett" ausgegeben werden, am besten in einer *form_alert()*-Box.

Schreiben ins Klemmbrett

Das Schreiben ins Klemmbrett funktioniert im Prinzip genauso wie das Lesen.
Als erstes sollte man alle Dateien aus dem Klemmbrettordner entfernen. Dazu holt man sich den Pfadnamen mit *scrp_read()* aus dem GEM-Speicher und löscht reihenweise mit Hilfe der Funktionen *Fsfirst()* und *Fsnext()* sowie *remove()* alle Dateien in diesem Pfad.

Dann hängt man an den Pfadnamen den gewünschten SCRAP.*-Dateinamen an. Unter diesem Namen speichert man anschließend seine Daten einfach ab. Dabei sollte man, wie schon gesagt, darauf achten, möglichst immer irgendein Standardformat zu benutzen, da sonst der Datenaustausch zwischen verschiedenen Applikationen nicht mehr möglich ist.

Die Scrap-Bibliothek

Um dem Programmierer den Umgang mit dem Klemmbrett zu vereinfachen, habe ich eine kleine Bibliothek aus vier Funktionen geschrieben, die einem die wichtigsten Arbeitsgänge mit dem Klemmbrett abnehmen können.

Sie können mit einem beliebigen C-Compiler compiliert werden und als Bibliothek dann zum eigenen Programm hinzugelinkt werden. Die Header-Datei enthält alle nötigen Vorwärts-Deklarationen, die zum korrekten Compilieren notwendig sind.

Hier eine kurze Beschreibung für jede dieser Funktionen:

Word Scrp_init()
Die Funktion *scrp_init(char *pfad)* dient zum Initialisieren des Klemmbretts, falls es noch nicht gesetzt ist. Die Applikation sollte diese Funktion in der eigenen Initialisierungsphase verwenden. Als Pfad benutzt man am besten einen vorgegebenen Standardpfad wie C:\CLIPBRD oder A:\CLIPBRD. Das sollte auch für den Benutzer änderbar sein. *Scrp_init()* liefert ein Wort zurück, das entweder 1 für keinen oder 0 für einen aufgetretenen Fehler enthält. Es empfgiehlt sich, diese Funktion auf jeden Fall aufzurufen, bevor eine der drei nachfolgenden angesprochen wird. Die Möglichkeit mit der Environment-Variablen CLIPBRD ist in dieser Funktion nicht verwirklicht, da ich diese Funktion auch in Accessories verwenden will, und Lattice-C bei Accessories keine Envi-

ronment-Variablen-Auswertung zuläßt (meines Wissens sind beim Start der Accessories auch noch keine vorhanden oder gesetzt).

Void Scrp_clear()
Diese Funktion löscht einfach alle Dateien in dem durch den GEM-Puffer beschriebenen Ordner. Sie sollte immer aufgerufen werden, bevor neue Daten ins Klemmbrett gelegt werden.

Word Scrp_length()
Die Funktion *scrp_length()* liefert die Summe aller Dateilängen im Klemmbrettordner in Kilobytes als Wort zurück, damit also die momentane Größe des Klemmbretts.

Word Scrp_find(byte *ext, byte *name)
Scrp_find() ist wohl die wichtigste Funktion. Sie liefert zu einer gegebenen Extension (z.B. "TXT" für einen ASCII-Text) den vollständigen Dateinamen der zugehörigen Klemmbrettdatei mit Pfad zurück, unter dem sie dann gefunden werden kann. Der Rückgabewert gibt an, wieviele Dateien unter diesem Suchkriterium (z.B. bei Wildcards) gefunden wurden; es wird jedoch immer nur der erste gefundene im name-Feld zurückgegeben. Das name-Feld sollte groß genug sein, um einen kompletten Pfad und Dateinamen aufnehmen zu können. Natürlich läßt sich diese Funktion auch benutzen, um den Dateinamen mit dem kompletten Pfad für die schreibende Datei herauszufinden. In dem Fall ist natürlich der Rückgabewert bedeutungslos, da diese Datei ja natürlich noch nicht existiert.

Es wäre zu begrüßen, wenn noch mehr Programmierer ihre Programme mit einer Klemmbrettunterstützung ausstatten würden, da dies eigentlich keinen großen Mehraufwand mit sich bringt.



```
  1:  #include "stdio.h"
  2:  #include "stdlib.h"
  3:  #include "string.h"
  4:  #include "osbind.h"
  5:  #include "dos.h"
  6:  #include "portab.h"
  7:  #include "gemlib.h"
  8:
  9:
 10:
 11:  /* +----------------------------------------+ */
 12:  /* | KLEMMBRETT INITIALISIEREN              | */
 13:  /* +----------------------------------------+ */
 14:
 15:  word      scrp_init(pfad)
 16:   byte *pfad;
 17:
 18:  {
 19:   byte scrap[64];
 20:   long handle;
 21:
 22:   scrp_read(scrap);
 23:   if(scrap[0]==0)
 24:   {        strcpy(scrap,pfad);
 25:            scrp_write(scrap);
 26:            {        handle = Dcreate(scrap);
 27:                     if((handle<0)&&(handle!=-36))
 28:                     {        scrp_write("");
 29:                              form_error(1);
 30:                              return(0);
 31:                     }
 32:            }
 33:   }
 34:   return(1);
 35:  }
 36:
 37:
 38:
 39:  /* +----------------------------------------+ */
 40:  /* | DATEIEN IM SCRAP-DIRECTORY LÖSCHEN     | */
 41:  /* +----------------------------------------+ */
 42:
 43:  void      scrp_clear()
 44:
 45:  {         char xpfad[ FMSIZE],gesammtpfad[ FMSI-
ZE],
                    xname[ FMSIZE],scrap[256];
 46:   struct FILEINFO file;
 47:   int ret;
 48:
 49:   scrp_read(scrap);
 50:   strmfp(xpfad,scrap,"SCRAP.*");
 51:   ret=dfind(&file,xpfad,0);
 52:   while(ret==0L)
 53:   {        stcgfp(gesammtpfad,xpfad);
 54:            strmfp(xname,gesammtpfad,file.name);
 55:            remove(xname);
 56:            ret=dnext(&file);
 57:   }
 58:
```

```
 59:  }
 60:
 61:
 62:
 63:  /* +----------------------------------------+ */
 64:  /* | LÄNGE DES KLEMMBRETT-INHALTS ALS WORD IN
          KBYTES LIEFERN                          | */
 65:  /* +----------------------------------------+ */
 66:
 67:  word      scrp_length()
 68:
 69:  {
 70:   byte name[ FMSIZE], filename[ FMSIZE];
 71:   struct FILEINFO dta;
 72:   long length = 0;
 73:
 74:   scrp_read(filename);
 75:   strmfp(name,filename,"SCRAP.*");
 76:
 77:   if(!dfind(&dta,name,0))
 78:            do
 79:                     length += dta.size;
 80:            while(!dnext(&dta));
 81:   else
 82:            return(0);
 83:
 84:   return((word)((length+512)/1024));
 85:  }
 86:
 87:
 88:
 89:  /* +----------------------------------------+ */
 90:  /* | EINE ROUTINE, DIE ZU EINER GEGEBENEN
          EXTENSION DAS CLIPFILE LIEFERT          | */
 91:  /* +----------------------------------------+ */
 92:
 93:  word      scrp_find(extension,filename)
 94:   byte *extension,*filename;
 95:
 96:  {
 97:   byte datei[ FMSIZE];
 98:   byte scrap[256];
 99:   word c = 0;
100:   struct FILEINFO dta;
101:
102:   scrp_read(scrap);
103:   strcpy(datei,scrap);
104:   strcat(datei,"\\SCRAP.");
105:   strcat(datei,extension);
106:   if(!dfind(&dta,datei,0))
107:   {
108:            c++;
109:            strcpy(datei,scrap);
110:            strmfp(filename,datei,dta.name);
111:            while(!dnext(&dta)) c++;
112:   }
113:   else
114:            return(0);
115:   return(c);
116:  }
```

```
 1:  /* -------------------------------------------- */
 2:  /*                                              */
 3:  /* SCRAPLIB.H                                   */
 4:  /* -------------------------                    */
 5:  /* (c) 1990 by Don Chaos                        */
 6:  /*                                              */
 7:  /* Eric Böhnisch                                */
 8:  /* In den Beeten  80                            */
 9:  /* 7121 Ingersheim 1                            */
10:  /* Phone 07142 51661                            */
11:  /*                                              */
12:  /* Dieser Header ist in Verbindung mit der
        SCRAPLIB.BIN-Library                         */
13:  /* zu verwenden. Er enthält alle
        Definitionen für die einzelnen               */
14:  /* Funktionen.                                  */
15:  /*                                              */
16:  /* -------------------------------------------- */
17:
18:
19:
20:  void      scrp_clear();
21:  word      scrp_init(byte *);
22:  word      scrp_length();
23:  word      scrp_find(byte *, byte *);
```

REKURSION IN OMIKRON.BASIC

Ulrich Hirschmann

WAS IN ANDEREN SPRACHEN MÖGLICH IST, MUSS DOCH AUCH IN BASIC MÖGLICH SEIN. UND MIT EINEM KLEINEN TRICK IST ES DAS AUCH. IM RAHMEN EINES LOGO-SEMINARS KAMEN DIE MANNIGFALTIGEN VORZÜGE VON LOGO ZUR SPRACHE, DIE ES UNTER ANDEREM ERLAUBEN, AUF KÜRZESTE WEISE REKURSIVE PROGRAMME ZU SCHREIBEN. DIESES SEI IN EINER "PRIMITIVSPRACHE" WIE ZUM BEISPIEL BASIC NUR SCHWER MÖGLICH. UM ÄHNLICHE RESULTATE ZU ERZIELEN, MÜSSTE MAN SICH IN BASIC SEITENLANGER PROGRAMME BEDIENEN.

Diese Auffassung zu widerlegen, war für mich der Anstoß für meinen Versuch in OMIKRON-BASIC. Mit nur zwei weiteren Befehlen läßt sich das Kapitel der Rekursionen auch in BASIC abhandeln. Allerdings verlangt die BASIC-Lösung auch etwas mehr Gedankenaufwand vom Programmierer.

Wie dem Listing zu entnehmen ist, habe ich auf zwei beliebte LOGO-Beispiele zurückgegriffen. Schnee zeichnet, wie der Name schon sagt, Schneekristalle auf den Bildschirm, wobei die Anzahl der Radien variabel ist. Zweige liefert die bekannte Baumstruktur mit variablem Winkel. Und Show liefert interessante Grafiken aus sich drehenden Quadraten. Es sind aber durchaus auch andere Beispiele möglich und relativ leicht selbst zu erstellen.

Die implementierten LOGO-Befehle dienen nur dem Zweck der Demonstration. So treten Rundungsfehler auf, die sich auch nur in einem Demoprogramm vertreten lassen. Aber vielleicht weiß ja jemand bessere Algorithmen?

Da Omikron-BASIC sich nicht um die Verwaltung der Variablen bei Selbstaufrufen von Prozeduren kümmert, müssen wir das eben selbst erledigen. Zu diesem Zweck erhält jeder Aufruf der Prozedur einen eigenen Variablensatz durch den Index *Eb*. Bei diesem Verfahren ist die Rekursionstiefe nur durch den Speicherplatz begrenzt, da die Variablen als Array definiert sind. Doch nun zur Beschreibung des Listings: Nach der Initialisierung folgt das Hauptprogramm, welches eines der Beispiele aufruft. Der Prozedur Schnee werden folgende Parameter übergeben: Eb - das ist die Zählervariable für die Rekursionsebenen, *S(Eb)* - bestimmt die Länge der Anfangsstrecke, *N(Eb)* - die Anzahl der Radien. In der Prozedur *Zweige* wird durch *N(Eb)* der Winkel zwischen zwei Ästen übergeben. Die Übergabe des Wertes Null durch *Eb* beim Start der Prozeduren dient zum Zurücksetzen des Rekursionszählers. Zeile 63 enthält die Abbruchbedingung bei Unterschreiten einer Mindestlänge. In der nächsten Zeile befindet sich die Zählschleife für die Anzahl der zu zeichnenden Radien. Der Befehl *Fd(S(Eb))* zeichnet nun eine Gerade mit der Länge *S*. Ihre Richtung ist vom letzten Aufruf der Prozedur *Rt()* abhängig, die bei der Initialisierung auf 0 Grad (senkrecht nach oben) gesetzt wurde.

Vor dem Selbstaufruf der Prozedur müssen nun der Rekursionsebenenzähler erhöht und der Schleifenzähler *I(Eb)* zurückgesetzt werden. Das erledigt die Prozedur *Rec_up*. Ebenso werden nach dem Selbstaufruf der Ebenenzähler wieder erniedrigt und der Schleifenzähler wieder auf seinen vorigen Wert gebracht *(Rec_down)*. Bis zur Zeile 68 existieren beim Programmablauf lediglich eine bestimmte Anzahl von Schneeprozeduren, die bisher nur eine Gerade gezeichnet und sich dann selbst wieder aufgerufen haben. In Zeile 69 zeichnet zuerst die zuletzt erzeugte Prozedur eine Gerade rückwärts *[Bk(S(Eb))]*, und dann wird ein Winkel nach rechts vollzogen. Durch den Index *Eb* existiert praktisch für jede Rekursionsebene ein eigener Variablensatz.

Beim Selbstaufruf der Prozedur muß aber auf die bereits wertzugewiesenen Variablen zurückgegriffen werden. Deshalb erfolgt der Aufruf mit *S(Eb-1)*. Die Division durch 2 erniedrigt lediglich kontinuierlich die Streckenlänge. Das Beispiel Zweige schaut da durch den doppelten Selbstaufruf schon wesentlich wilder aus. Die Prozedur *Rechtecke* verdeutlicht die Wirkungsweise von Rekursionen noch einmal sehr anschaulich. Nur wird hier bei jedem Selbstaufruf die x-Koordinate verringer. Die LOGO-Befehle erklären sich mit ein wenig Trigonometrie von selbst. Nur sei nochmals darauf hingewiesen, daß sie nur für Demonstrationszwecke gedacht sind. Allerdings benötigt man mit diesen Befehlen, z.B. für die beliebte Analoguhr auf dem ST, nur maximal 20 Zeilen. Versuchen lohnt sich! Zum Schluß sei noch darauf hingewiesen, daß es natürlich nicht meine Absicht ist, BASIC als die Hochsprache schlechthin zu verteidigen. Speziell Rekursionen lassen sich mit anders strukturierten Sprachen sicher besser bearbeiten. Aber möglich ist es in BASIC eben doch.

```
  1:  '*********************************************
  2:  '* Rekursionen
  3:  '* in
  4:  '* OMIKRON-Basic
  5:  '*
  6:  '* Ulrich Hirschmann
  7:  '*
  8:  '*@ 02.1990,
  9:  '*(c) MAXON Computer GmbH 1990
 10:  '*********************************************
 11:  Init
 12:  '
 13:  REPEAT
 14:    L_Input(0,0,"Parameter eingeben :",W%L)
 15:    I%L(0)=0:L_Wa%L=0
 16:    PRINT @(2,0);"Ende: Winkel>90 oder [Help]"
 17:    CLS
 18:    Sety%L=250
 19:    Setx%L=250
 20:  '   Zweige(0,50,W)'     alternativ    ********
 21:    Schnee(0,50,W%L)'                  ********
 22:
 23:  ' Show'        ***    LOGO-Demo    ***
 24:   WHILE MOUSEBUT =0: WEND
 25:   IF MOUSEBUT =2 THEN HCOPY
 26:  UNTIL Immer%L
 27:  '
 28:  END
 29:  '************** Logoprogramm ****************
 30:  DEF PROC Show
 31:        FILL STYLE =1,0
 32:        Pe
 33:        Sety%L=200
 34:        PRINT CHR$(27);"p"
 35:        WHILE 1
 36:          Setx%L=600
 37:          Rt(0):Fd(0)
 38:          L_Wa%L=0
 39:          INPUT @(0,0);"Winkel :";I%L
 40:          PBOX 0,0 TO 640,400
 41:          Rechtecke(0,6,I%L)
 42:        WEND
 43:        Pd
 44:        PRINT CHR$(27);"q"
 45:  RETURN
 46:  DEF PROC Quadrat(St%L)
 47:         FOR I%L=1 TO 4
 48:           Fd(St%L):Rt(90)' **mit rt(33) probieren
 49:         NEXT I%L
 50:  RETURN
 51:  DEF PROC Rechtecke(Eb%L,S!(Eb%L),N%L(Eb%L))
 52:        IF S!(Eb%L)>200 THEN RETURN
 53:        Lt(N%L(Eb%L))
 54:        Quadrat(S!(Eb%L))
 55:        Setx%L=Setx%L-4
 56:        Rec_Up'        *** nächste Ebene
 57:        Rechtecke(Eb%L,S!(Eb%L-1)+3,N%L(Eb%L-1))
 58:        Rec_Down'      *** und zurück
 59:        Lt(N%L(Eb%L))
 60:        Quadrat(S!(Eb%L))
 61:        Setx%L=Setx%L-4
 62:  RETURN
 63:  DEF PROC Schnee(Eb%L,S!(Eb%L),N%L(Eb%L))'
           *** Eb setzt den Rekursionszähler zurück
 64:        IF S!(Eb%L)<4 THEN RETURN '   *** Abbruch
 65:        WHILE I%L(Eb%L)<N%L(Eb%L)'    *** Anzahl
                                  der gezeichneten Radien
 66:          Fd(S!(Eb%L))
 67:          Rec_Up'                *** nächste Ebene
 68:          Schnee(Eb%L,S!(Eb%L-1)/2,N%L(Eb%L-1))
 69:          Rec_Down'              *** und zurück
 70:          Bk(S!(Eb%L))
 71:          Rt(360/N%L(Eb%L))
 72:        WEND
 73:  RETURN
 74:  '
 75:  DEF PROC Zweige(Eb%L,S!(Eb%L),N%L(Eb%L))
 76:        IF S!(Eb%L)<10 THEN RETURN ' *** minimale
                                          Streckenlänge
 77:        Lt(N%L(Eb%L)):Fd(S!(Eb%L))
 78:        Rec_Up
 79:        Zweige(Eb%L,S!(Eb%L-1)/1.2,N%L(Eb%L-1))
 80:        Rec_Down
 81:        Bk(S!(Eb%L))
 82:        Rt(N%L(Eb%L)*2):Fd(S!(Eb%L))
 83:        Rec_Up
 84:        Zweige(Eb%L,S!(Eb%L-1)/1.2,N%L(Eb%L-1))
 85:        Rec_Down
 86:        Bk(S!(Eb%L)):Lt(N%L(Eb%L))
 87:  RETURN
 88:  '
 89:  '*********************************************
 90:  '*********** Logobefehlsdefinitionen ***********
 91:  '*********************************************
 92:  '
 93:  DEF PROC L_Input(L_X%L,L_Y%L,L_Text$,R W%L)
 94:         PRINT CHR$(27);"e"
 95:         -J: INPUT @(L_X%L,L_Y%L);L_Text$;W%L
 96:         IF W%L=0 THEN GOTO -J%L
 97:         PRINT CHR$(27);"f"
 98:  RETURN
 99:  DEF PROC Warte: WHILE INKEY$ ="": WEND : RETURN
100:  DEF PROC Rec_Up:Eb%L=Eb%L+1:I%L(Eb%L)=0: RETURN
101:  DEF PROC Rec_Down:Eb%L=Eb%L-1:
             I%L(Eb%L)=I%L(Eb%L)+1: RETURN
102:  DEF PROC Pu: LINE COLOR =0: RETURN ' *** Pen up
103:  DEF PROC Pd: LINE COLOR =1: RETURN ' *** Pen dn
104:  DEF PROC Px: MODE =3: RETURN ' *** Pen XOR
105:  DEF PROC Pe: MODE =1: LINE COLOR =0: RETURN
                                   ' *** Pen erease
106:  '
107:  DEF PROC Init
108:    ON HELP GOSUB L_Ende
109:    CLS : DEG : PRINT CHR$(27);"f": CLIP 0,0,640,
                                               400
110:    Rt(0):Fd(0)
111:    DIM Eb%L(999),I%L(999),S!(999),N%L(999)
112:    MODE =1:Pd
113:  RETURN
114:  '
115: -L_Ende: END
116:  '
117:  DEF PROC Rt(Winkel%L)'          *** right turn
118:    L_W1%L=Winkel%L+L_Wa%L
119:    L_Wa%L=L_W1%L
120:    L_W%L=L_W1%L-90
121:  RETURN
122:  '
123:  DEF PROC Lt(Winkel%L)'          *** left turn
124:    L_W1%L=-Winkel%L+L_Wa%L
125:    L_Wa%L=L_W1%L
126:    L_W%L=L_W1%L-90
127:  RETURN
128:  '
129:  DEF PROC Fd(Strecke%L)'         *** forward
130:         X1%L=Setx%L:Y1%L=Sety%L
131:         X2%L=Strecke%L* COS(L_W%L)+Setx%L
132:         Y2%L=Strecke%L* SIN(L_W%L)+Sety%L
133:         DRAW X1%L,Y1%L TO X2%L,Y2%L
134:         Setx%L=X2%L:Sety%L=Y2%L
135:  RETURN
136:  '
137:  DEF PROC Bk(Strecke%L)'         *** backward
138:         Setxb%L=Setx%L:Setyb%L=Sety%L
139:         X1%L=Setxb%L:Y1%L=Setyb%L
140:         X2%L=Strecke%L* COS(L_W%L-180)+Setxb%L
141:         Y2%L=Strecke%L* SIN(L_W%L-180)+Setyb%L
142:         DRAW X1%L,Y1%L TO X2%L,Y2%L
143:         Setxb%L=X2%L:Setyb%L=Y2%L
144:         Setx%L=X2%L:Sety%L=Y2%L
145:  RETURN
```

DATEN AUF DEM HOLZWEG

Michael Krusemark

ZUNÄCHST EINMAL ZU DEM ZWECK DIESES PROGRAMMS. ES DIENT ZUR DATENÜBERTRAGUNG ZWISCHEN ZWEI STS. MANCHER WIRD SICH NUN DENKEN, DAS SEI DOCH NICHTS BESONDERES. WER ABER EIN LÄNGERES KABEL BENÖTIGT, WIRD FESTSTELLEN, DASS DIE ÜBERTRAGUNG MIT 2400 BAUD ZWAR GANZ GUT FUNKTIONIERT, ABER ZU LANGSAM IST, 4800 BAUD WIEDERUM ZU SCHNELL UND DAMIT NICHT DIE ERFORDERLICHE SICHERHEIT BIETET.

In meiner Not habe ich versucht, etwas dazwischen zu erreichen. Das serielle Kabel besitzt nicht nur Sende- und Datenleitungen, sondern auch sogenannte Handshake-Leitungen. Im Normalfall sollen sie eine sichere Datenübertragung gewährleisten. Meine Idee war nun folgende: Ich mißbrauche die Handshake-Leitungen, die man (fast) beliebig auf High oder auf Low setzen kann, ohne sich an irgendwelche Taktraten zu halten. Über die Leitung RTS/CTS wird dem Empfänger gemeldet, daß nun Daten kommen, über DCD/DTR werden dann die eigentlichen Daten übertragen. Es gibt zwei Möglichkeiten festzustellen, ob Daten anliegen. Die erste ist das ständige Abfragen der entsprechenden Hardware-Register im MFP (Polling), die zweite das Auslösen eines Interrupts, wenn CTS auf High geht. Diese Methode ist wesentlich eleganter, da sie einen Multitasking-Betrieb erlaubt und somit nicht kostbare Rechenzeit verloren geht. Daher habe ich mich für die letzte Lösung entschieden.

Die Entwickler des ST haben es dem Programmierer aber nicht so einfach gemacht, daß ein Interrupt ausgelöst wird, wenn ein High an CTS anliegt, nein, es muß zu einer Flanke von High auf Low kommen (kann im MFP aber festgelegt werden). Nun alles der Reihe nach: Wenn der Sender eine Flanke am CTS-Eingang des Empfängers erzeugt, wird ein Interrupt (CTS-Interrupt, MFP-Int.-Nr. 2) ausgelöst. Während des Interrupts wird an Pin 8 (DCD) des seriellen Ports abgefragt, ob eine logische 1 oder 0 empfangen wird. Das Bit wird in das bisherige Byte eingeflickt und dann, wenn das Byte noch nicht komplett empfangen wurde, zwischengespeichert (inpchar), ansonsten wird das Zeichen in den Buffer geschrieben und alles für das nächste Zeichen vorbereitet. Das Empfangen von Daten wäre hiermit erledigt. Nun zum Senden: Auch das geschieht über einen Interrupt.

Von den 4 Timern steht einer, nämlich der Timer A, dem Programmierer frei zur Verfügung. Mit ihm werden die einzelnen Bits durch Setzen oder Löschen des DTR gesendet (die Gegenstation empfängt somit ein DCD). Die zu sendenden Bytes werden aus dem Sendepuffer (gleich dem Empfangspuffer) geholt. Nach jeweils 8 übermittelten Bits wird ein neues Zeichen aus dem Puffer gelesen. Wie oben schon erwähnt, muß eine Flanke ausgelöst werden. Das geschieht bei jedem zweiten Timer-Interrupt. In den anderen werden die Zustände des DTR-Ausgangs festgelegt. Bei dem Versuch, beides in einem Interrupt zu erledigen, gab es Datensalat, da es scheinbar doch eine gewisse Zeit dauert, bis die Pegel an den Ausgängen erreicht sind.

Den Leser, der bis hierher durchgehalten hat, kann ich beruhigen, das gröbste haben wir hinter uns. Die Zeichen, die empfangen oder gesendet werden, stehen in einem Ringbuffer. Das heißt, daß eine Schreib- und Lesemarke existiert. Zu jedem Buffer gehören noch diverse andere Angaben, die von ihrem Namen her selbsterklärend sind. Die Routinen *auxout* und *auxin* dienen zum Beschreiben und Lesen der Buffer. Sie sind, wie alle anderen Routinen und die Verwaltung der Buffer, dem Betriebsystem nachempfunden. Die Bezeichnung der Routinen und Symbole lassen ihre Funktion schon erkennen. Wer genaueres erfahren möchte, sollte sich mit dem heftig dokumentierten Source befassen. Die kurze Demo am Anfang des Programms, von dem Rest durch '-' getrennt, dient zur Veranschaulichung der Routinen. Das Programm soll nur eine Sammlung von Unterprogrammen sein, die eine neue Dimension der Datenübertragung via Modemkabel darstellen. Die Routinen sind sehr offen angelegt, das heißt es ist ohne viel Änderung des Programmtextes weit konfigurierbar. Die Geschwindigkeit der Datenübertragung wird dann von den Konstanten *speed* und *delay* festgelegt. Die Größe der Buffer stehen in *max_out_size* sowie *max_in_size*.

Nun ist Ihre Initiative gefragt. Sie haben alle Routinen, die eine Datenübertragung im Multitaskingbetrieb mit Ringbuffern ermöglichen. Da wäre zum Beispiel die Möglichkeit eines Low-Cost-Netzwerkes oder der gleichzeitige Datentransfer mit dem Betriebssystem und diesen Routinen oder... Besonders jetzt, da viele Anwender neben ihrem 'kleinen' ST einen Mega besitzen, sollte man diesen nicht in einer Ecke vergammeln lassen. Er könnte auch als Coprozessor oder zu einer absolut resetfesten RAM-Disk dienen.

Allerdings muß darauf hingewiesen werden, daß der Datentransfer auch seine Nachteile hat:

1) Wer eine hohe Baudrate erreichen möchte, muß in Kauf nehmen, daß sich das in der Rechengeschwindigkeit bei Sender und Empfänger bemerkbar macht.

2) Wird während der Übertragung auf die Diskette zugegriffen, kann es vorkommen, daß die Fehlermeldung 'Daten auf Disk X: defekt?' erscheint. Normalerweise hilft hier, das Ende der Übertragung abzuwarten und dann den Diskettenzugriff zu wiederholen.

3) werden die Daten nicht schnell genug aus dem Buffer gelesen, beißt sich der Hund in den Schwanz. Das ist so zu verstehen, daß die Schreibmarke die Lesemarke überholt, was zu Datenverlust führt.

Um dieses mächtige Werkzeug in Gang zu bringen, ist es lediglich nötig, sich ein serielles Kabel zu basteln. Um eine hohe Baudrate zu erreichen, rate ich, abgeschirmte Kabel zu verwenden. Die Pinbelegung können sie der Tabelle entnehmen.

Pinbelegung:

```
RTS  4 <——> 5 CTS
CTS  5 <——> 4 RTS
DCD  8 <——>20 DTR
DTR 20 <——> 8 DCD
GR   7 <——> 7 GR (Signal-
                  masse)
```

Pin 1 dient als Schutzerde und sollte nur an einem Anschluß mit der Abschirmung verbunden sein. Braucht man zwar nicht unbedingt, ist aber besser so.

Wer gleichzeitig noch seine Daten mit Hilfe des Betriebssystems übertragen möchte, muß folgende Verbindung noch hergestellt werden:

```
RXD 2 <——> 3 TXD
TXD 3 <——> 2 RXD
```

Literatur:

ATARI ST Profibuch, H.-D. Jankowski/ J.F. Reschke/ D.Rabich, 7.Auflage, Sybex-Verlag 1989 S.700 ff



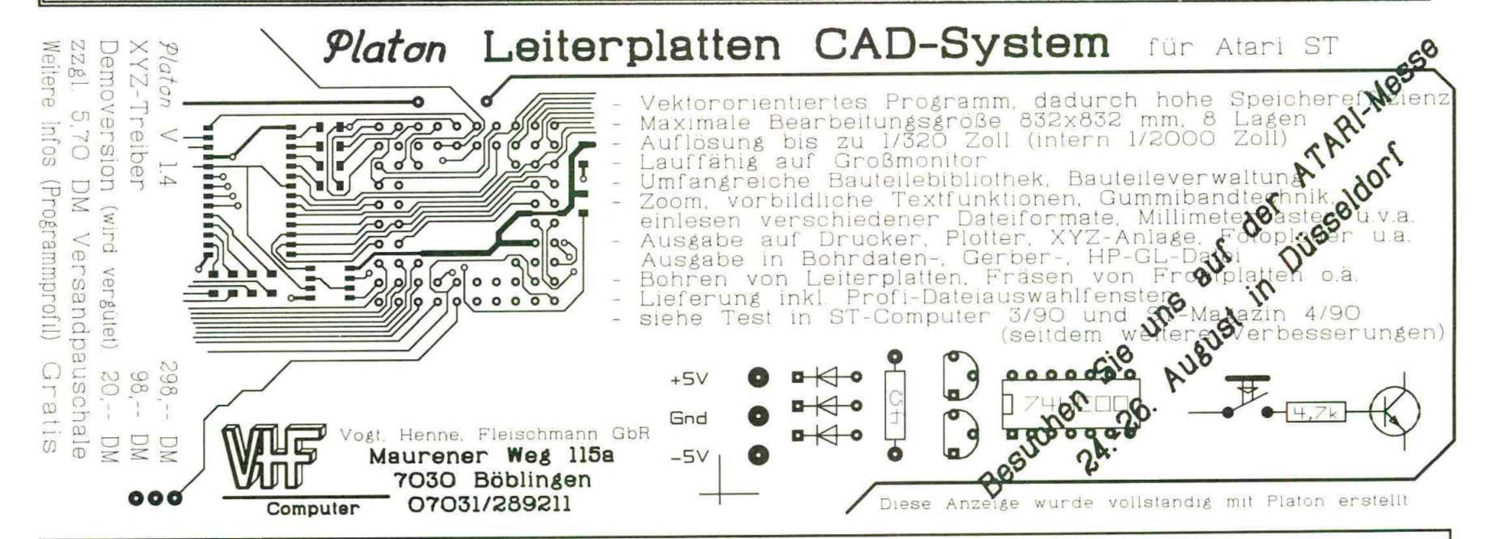
```
 1:  ;*************************************************
 2:  ;* Routinen zu Datenübertragung zwischen 2 Atari
 3:  ;* ST unter Verwendung der Handshake Leitungen
 4:  ;* Autor: Michael Krusemark
 5:  ;*
 6:  ;*
 7:  ;*         STCOM V.1.30 (27.1.1990)
 8:  ;*         (c) MAXON Computer GmbH 1990
 9:  ;*************************************************
10:
11:  MFPbase        equ $fffa00
12:  MFPgpio        equ $fffa01
13:  tadr           equ $1f
14:  tacr           equ $19
15:  isra           equ $f
16:  PSGselect      equ $ff8800
17:  Xbtimer        equ $1f
18:  Jdisint        equ $1a
19:  Setexec        equ 5
20:  Cconws         equ 9
21:  Cconis         equ 11
22:  Cconout        equ 2
23:  Cconin         equ 1
24:  Bcostat        equ 8
25:  XBIOS          equ 14
26:  BIOS           equ 13
27:  GEMDOS         equ 1
28:  speed          equ 20    ; < Das kann
29:  delay          equ 7     ; < alles
30:  max_out_size             equ 4096   ; < verändert
31:  max_inp_size             equ 4096   ; < werden
32:
33:                 bsr.s    initial     ;initialisieren
34:  ;-----------------------------------------------
35:                 pea      logo(pc)
36:                 move.w   #Cconws,-(sp)
37:                 trap     #GEMDOS
38:                 addq.l   #6,sp
39:  ewig:          move.w   #Cconis,-(sp)    ;Zeichen
                                     in Tastaturbuffer
40:                 trap     #GEMDOS
41:                 addq.l   #2,sp
42:                 tst.w    d0
43:                 bne.s    in_buffer   ;ja, -->
44:                 bsr      auxin       ;Zeichen
                                          von RS232
45:                 tst.w    d0          ;eins da?
46:                 bmi.s    ewig        ;nein, -->
47:                 cmpi.w   #27,d0      ;Escape
48:                 beq.s    aus         ;fertig
49:                 move.w   d0,-(sp)     ;Zeichen
50:                 move.w   #Cconout,-(sp)  ;ausgeben
51:                 trap     #GEMDOS
52:                 addq.l   #4,sp
53:                 bra.s    ewig         ;endlos
54:
55:  in_buffer:     move.w   #Cconin,-(sp)   ;Zeichen
                                  aus Tastaturbuffer
56:                 trap     #GEMDOS      ;holen
57:                 addq.l   #2,sp
58:                 cmpi.w   #27,d0       ;Escape?
59:                 beq.s    aus          ;ja, -->
60:                 bsr      auxout       ;Zeichen senden
61:                 bra.s    ewig         ;endlos
62:  ;-----------------------------------------------
63:  aus:           bsr.s    deinstall    ;deinstallieren
64:                 clr.w    -(sp)        ;Pterm
65:                 trap     #GEMDOS
66:
67:  initial:       clr.l    inpchar      ;Zwischenspei-
                                           cher löschen
68:
69:                 pea      timerA(pc) ;Timer A Vektor
                                           verbiegen
70:                 move.w   #77,-(sp)    ;Execptionvek-
                                           tor 77
71:                 move.w   #Setexec,-(sp)
72:                 trap     #BIOS
73:                 addq.l   #8,sp
74:                 move.l   d0,old_timer ;alten
                                  ;Vektor speichern
76:                 pea      cts(pc)      ;CTS-Vektor
                                           verbiegen
77:                 move.w   #66,-(sp)    ;Exception 66
78:                 move.w   #Setexec,-(sp)
79:                 trap     #BIOS
80:                 addq.l   #8,sp
81:                 move.l   d0,old_cts ;alten Vektor
                                           speichern
82:                 rts
83:
84:  deinstall:     move.w   #13,-(sp)    ;Timer aus
85:                 move.w   #Jdisint,-(sp)
86:                 trap     #XBIOS
87:                 addq.l   #4,sp
88:
89:                 move.l   old_timer(pc),-(sp)
90:                 move.w   #77,-(sp)    ;Timervektor
                                           restaurieren
91:                 move.w   #Setexec,-(sp)          →
```

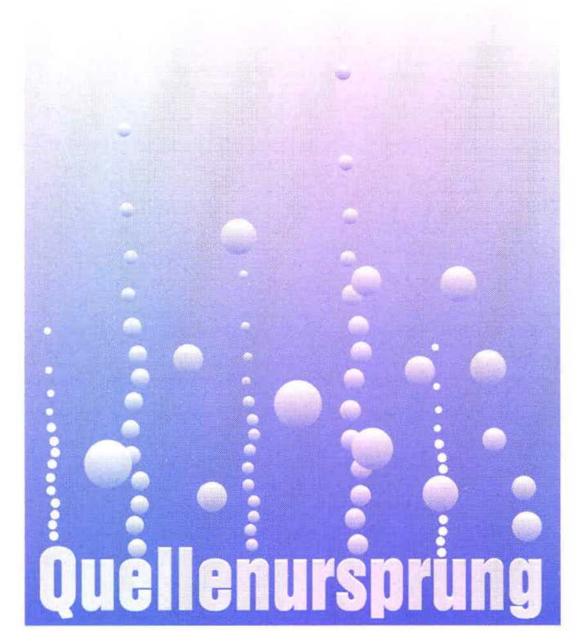
```
 92:                trap     #BIOS
 93:                addq.l   #8,sp
 94:
 95:                move.l   old_cts(pc),-(sp)
 96:                move.w   #66,-(sp)  ;CTS-Vektor
                                        restaurieren
 97:                move.w   #Setexec,-(sp)
 98:                trap     #BIOS
 99:                addq.l   #8,sp
100:                rts
101:
102:  cts:          movem.l  d0/d1/a0,-(sp)
103:                move.w   sr,-(sp)
104:                ori.w    #$0700,sr      ;IPL 7
105:                move.b   inpbits(pc),d1  ;Bits zu
                                              lesen
106:                bne.s    get_bit
107:                move.b   #8,d1
108:  get_bit:      move.b   inpchar(pc),d0
109:                lsl.b    #1,d0        ;nach links
110:                btst     #1,MFPgpio ;DCD gesetzt
111:                bne.s    set_bit      ;ja> Bit setzen
112:                bset     #0,d0        ;nein, setzen
113:  set_bit:      bchg     #0,d0        ;setzen/löschen
114:                subq.b   #1,d1        ;ein Bit weniger
115:                bne.s    ade
116:                movea.l  inpbuf(pc),a0 ;Bufferadr.
117:                adda.l   inphd(pc),a0  ;+Head
118:                cmpa.l   inptop(pc),a0 ;unter Top?
119:                bcs.s    under_top1     ;ja -->
120:                movea.l  inpbuf(pc),a0 ;Schreibposi-
                                     tion=Bufferadresse
121:                clr.l    inphd        ;Head ist null
122:  under_top1:   move.b   d0,(a0)      ;schreiben
123:                addq.l   #1,inphd     ;nächste Pos.
124:                clr.b    d0           ;Zeichen löschen
125:                move.b   #8,d1
126:                addq.l   #1,inplen
127:  ade:          move.b   d1,inpbits ;schreiben
128:                move.b   d0,inpchar ;speichern
129:                move.w   (sp)+,sr
130:                movem.l  (sp)+,d0/d1/a0
131:                move.l   old_cts(pc),-(sp)
132:                rts                   ;zur alten
                                           Routine
133:
134:  outstat:      move.l   outlen(pc),d0    ;Anzahl
                                 der Zeichen in Buffer
135:                rts
136:
137:  auxout:       move.w   d0,-(sp)
138:                tst.b    power
139:                bne      is_space
140:                st       power
141:
142:                pea      timerA(pc) ;Timer A
                                        installieren
143:                move.w   #speed,-(sp)     ;Data
144:                move.w   #delay,-(sp)     ;Delay
145:                clr.w    -(sp)
146:                move.w   #Xbtimer,-(sp)
147:                trap     #XBIOS
148:                lea.l    $c(sp),sp
149:
150:  is_space:     move.l   outsize(pc),d1
151:  test_buffer:  bsr.s    outstat
152:                cmp.l    d1,d0        ;warten, bis
                                  Platz im Buffer ist
153:                bhi.s    is_space     ;nochmal testen
154:                move.w   (sp)+,d0
155:                movea.l  outbuf(pc),a0 ;Bufferadr.
156:                adda.l   outhd(pc),a0  ;+ Head
157:                cmpa.l   outtop(pc),a0 ;unter Top?
158:                bcs.s    under_top2     ;ja -->
159:                movea.l  outbuf(pc),a0 ;Schreibposi-
                                     tion=Bufferadresse
160:                clr.l    outhd        ;Head ist null
161:  under_top2:   move.b   d0,(a0)      ;schreiben
162:                addq.l   #1,outhd     ;nächste Pos.
163:                addq.l   #1,outlen
164:                rts
165:
166:  timerA:       movem.l  d0/d1/a0/a1,-(sp)
167:                ori.w    #$0700,sr   ;IPL 7
168:                lea.l    MFPbase,a1
169:                clr.b    tacr(a1)     ;Interrupt aus
170:                tst.b    flag         ;Flanke erzeugen
                                          oder Bit senden
171:                beq      flanke       ;0 --> Flanke
                                          erzeugen
172:
173:                tst.b    outbits      ;noch Bits?
174:                bne.s    transmit     ;ja --> senden
175:
176:                move.b   #8,outbits ;noch 8 senden
177:                movea.l  outbuf(pc),a0 ;Bufferadr.
178:                adda.l   outtl(pc),a0   ;+ Tail
179:                cmpa.l   outtop(pc),a0 ;unter Top
180:                bcs.s    under_top3      ;ja -->
181:                movea.l  outbuf(pc),a0 ;Leseposíti-
                                      on=Bufferadresse
182:                clr.l    outtl        ;Tail ist null
183:  under_top3:   move.b   (a0),outchar  ;schreiben
184:                addq.l   #1,outtl     ;nächste Pos.
185:                subq.l   #1,outlen
186:
187:  transmit:     move.b   outchar(pc),d1
188:                pea      weiter(pc) ;zum bedingten
                                        Routinensprung
189:                lsl.b    #1,d1        ;oberstes Bit
                                          in CARRY
190:                bcs      set_dtr      ;Carry gesetzt
                                          DTR=1
191:                bcc      clr_dtr      ;Carry gelöscht
                                          DTR=0
192:  weiter:       bsr      set_rts      ;für Flanke
193:                subi.b   #1,outbits ;ein Bit weniger
194:                move.b   d1,outchar ;speichern
195:
196:  fertig:       move.b   #speed,tadr(a1) ;Interrupt
                                        initialisieren
197:                move.b   #delay,tacr(a1) ;und somit
                                           einschalten
198:  switch_out:   not.b    flag         ;Flag ändern
199:                bclr     #5,isra(a1)      ;fertig
200:                movem.l  (sp)+,d0/d1/a0/a1
201:                rte
202:
203:  flanke:       bsr.s    clr_rts
204:                tst.l    outlen       ;noch Zeichen
                                          in Buffer
205:                bne      fertig       ;ja, -->
206:                clr.b    power        ;sonst aus!
207:                clr.b    tacr(a1)     ;Sender aus
208:                bra.s    switch_out
209:
210:  inpstat:      move.l   inplen(pc),d0    ;Anzahl
                                            der Zeichen
211:                rts
212:
213:  auxin:        bsr.s    inpstat      ;Zeichen?
214:                tst.l    d0
215:                beq.s    kein_zeichen      ;nein -->
216:                movea.l  inpbuf(pc),a0     ;Bufferadr.
217:                adda.l   inptl(pc),a0      ;+ Tail
218:                cmpa.l   inptop(pc),a0     ;unter Top?
219:                bcs.s    under_top4        ;ja -->
220:                movea.l  inpbuf(pc),a0     ;Lesepos.
                                        =Bufferadresse
221:                clr.l    inptl        ;Tail ist null
222:  under_top4:   move.b   (a0),d0      ;Zeichen lesen
223:                addq.l   #1,inptl     ;nächste Pos.
224:                ext.w    d0           ;auf W erweitern
225:                subq.l   #1,inplen
226:                rts
227:  kein_zeichen:          move.w            #-1,d0
                                          ;nix in Buffer
228:                rts
229:
230:  set_dtr:      bsr.s    portA        ;bei folgenden
231:                bset     #4,d0        ;Routinen kann
232:                move.b   d0,2(a0)     ;ich mir eine
233:                rts                   ;Doku sparen
234:
235:  clr_dtr:      bsr.s    portA
236:                bclr     #4,d0
237:                move.b   d0,2(a0)
238:                rts
239:
240:  set_rts:      bsr.s    portA
241:                bset     #3,d0
242:                move.b   d0,2(a0)
243:                rts
244:
245:  clr_rts:      bsr.s    portA
```




```
246:            bclr    #3,d0
247:            move.b  d0,2(a0)
248:            rts
249:
250: portA:     lea.l   PSGselect,a0    ;PSGselect
251:            move.b  #14,(a0)        ;Port A
252:            move.b  (a0),d0         ;lesen
253:            rts
254:
255:            .DATA
256: outbuf:    .DC.l out_buffer
257: inpbuf:    .DC.l inp_buffer
258: outsize:   .DC.l max_out_size
259: inp_size:  .DC.l max_inp_size
260: outtop:    .DC.l out_buffer+max_out_size
261: inptop:    .DC.l inp_buffer+max_inp_size
262: ;---------------------------
263: logo:      .DC.b 27,'E',27,'v',0     ; Clear
264: ;---------------------------        Home, Wrap Line
265:            .BSS
```

```
266: old_cts:     .DS.l 1  ;Speicherung der
267: old_timer:   .DS.l 1  ;Vektoren
268:
269: inplen:      .DS.l 1  ;Länge
270: inphd:       .DS.l 1  ;Head
271: inptl:       .DS.l 1  ;Tail
272:
273: inpchar:     .DS.b 1  ;das sind
274: outchar:     .DS.b 1  ;alles
275: inpbits:     .DS.b 1  ;Zwischenspeicher
276: outbits:     .DS.b 1  ;zur 'Konstruktion' der
277:                       ;Bytes
278: outlen:      .DS.l 1  ;Länge
279: outhd:       .DS.l 1  ;Head
280: outtl:       .DS.l 1  ;Tail
281:
282: flag:        .DS.b 1  ;RTS oder Bit senden
283: power:       .DS.b 1  ;Senderschalter ein/aus
284: out_buffer:  .DS.b max_inp_size ;Empfangsbuffer
285: inp_buffer:  .DS.b max_out_size ;Sendebuffer
```



Resource-Formate

Geht es Ihnen als GEM-Programmierer nicht auch so, daß Sie schon recht lange mit Resource-Dateien arbeiten und eigentlich schon immer mal wissen wollten, was dahinter steckt? Oder wollten Sie nicht schon immer einmal wissen, was die Routine rsrc_load() so alles an Arbeit verrichtet? Leider findet man in der gängigen Literatur relativ wenig Informationen darüber, oder wissen Sie beispielsweise, für was man die Free-Strings und -Images verwenden kann und wie man sie anspricht?. Deshalb möchte ich Sie in diesem Monat damit beschäftigen, was als Ausgabe aus einem Resource-Construction-Set herauskommt - quasi als Hintergrund-Information zu dem Test der Resource-Construction-Sets in der letzten Ausgabe der ST-Computer...

Allgemeines

Auch wenn das Wissen um das Format einer solchen Datei nicht unbedingt für das Programmieren von GEM-Programmen wichtig ist, müssen wir doch alle zugeben, daß wir ein wenig neugierig sind. Außerdem werden wir sehen, daß uns das Wissen auf ein paar Dinge stoßen läßt, über die man sich ab und an wundert. Wenn Sie das Format verstanden haben, können Sie sogar eigene Daten in eine RSC-Datei verpacken oder mit einer eigenen Routine Resource-Dateien laden. Obwohl gerade dieses Format ein umfangreiches Verständnis der Objektstruktur von Dialogboxen und ähnlichem voraussetzt, möchte ich dennoch die dabei verwendeten Strukturen nicht in allen Einzelheiten besprechen, da dies sicherlich den Rahmen sprengen würde.

Mit Köpfchen voran

Das A und O einer Resource-Datei ist der sogenannte Header (Kopf). Dieser Header beinhaltet die Information, wo die unterschiedlichen Datenblöcke zu finden sind. Schauen wir uns also den Header an:

```
typedef struct rshdr{
    int     rsh_vrsn;          /* Versionsnummer */
    int     rsh_object;        /* OBJECT-Adresse */
    int     rsh_tedinfo;       /* TEDINFO-Adresse */
    int     rsh_iconblk;       /* ICONBLK-Adresse */
    int     rsh_bitblk;        /* BITBLK-Adresse */
    int     rsh_frstr;         /* Freestring-Index-Tabelle */
    int     rsh_string;        /* String-Adresse */
    int     rsh_imdata;        /* Bilddaten */
    int     rsh_frimg;         /* Freeimage-Index-Tabelle */
    int     rsh_trindex;       /* Baumindex-Tabelle */
    int     rsh_nobs;          /* Anzahl der Objekte */
    int     rsh_ntree;         /* Anzahl der Bäume */
    int     rsh_nted;          /* Anzahl der Tedinfos */
    int     rsh_nib;           /* Anzahl der Iconblks */
    int     rsh_nbb;           /* Anzahl der Bitblks  */
    int     rsh_nstring;       /* Anzahl der Strings  */
    int     rsh_nimages;       /* Anzahl der Bilder */
    int     rsh_rssize;        /* Länge der Resource-Datei */
}RSHDR;
```

Listing 1: Der Recource-Header

Eigentlich liegt der Großteil des Geheimnisses allein in dem **R**e**S**ource**H**ea**D**e**R**, deshalb werde ich ihn Stück für Stück interpretieren. Dabei bitte ich Sie, daß Sie sich die Abbildung 1 anschauen. Dort finden Sie eine Resource-Datei mit Adressen ausgegeben, somit können wir also die Theorie gleich mit der Praxis vergleichen.

Versionsnummern

Beginnen wir, wie so oft, am Anfang, wo wir als ersten Eintrag im Header der Resource-Datei die Versionsnummer finden, und schauen in Abbildung 1 nach, was dort steht. Sie wundern sich über die 0 in den beiden Bytes ? Ich muß zugeben, daß mich das auch wundert. Scheinbar wird diese Versionsnummer bei den RCS von Digital Research ignoriert. Allerdings habe ich festgestellt, daß eine einmal an diese Stelle geschriebene Zahl erhalten bleibt, auch wenn man die Datei bearbeitet und später wieder neu abspeichert. Das gilt aber nur für die RCS von Digital Research - das Kuma Resource sowie Wercs schreiben an diese Stelle eine '00 01' und überschreiben damit eine dort abgelegte Zahl - schade. Kommen wir nun aber zum Aufbau der Resource-Datei.

Kurze Zeiger

Eigentlich kann man gar nicht von einem generellen Aufbau einer Resource-Datei sprechen, da diese variieren kann. Festgelegt ist eigentlich nur, daß die Resource-

Datei einen Header besitzt, in dem enthalten ist, wo wiederum die einzelnen Datenblöcke innerhalb dieser Datei zu finden sind. Das bedeutet, daß man in dem RSHDR Zeiger findet, die die Position eines Datenblocks innerhalb der Datei ausmachen. Und damit fängt das Problem an, denn leider konnten sich die Programmierer des GEM wohl nicht vorstellen, daß eine Resource-Datei vielleicht irgendwann einmal länger als 64 KBytes werden könnte (ich vermute, daß dies wie so oft an der allseits geliebten 64K-Segmentierung unseres MS-DOSe-Intel-Prozessors liegt...). Daher haben sie - man muß ja Speicherplatz sparen - dem Resource-Header nur 2-Byte-Zeiger spendiert und nicht, wie zu wünschen gewesen wäre, 4-Byte-Zeiger. Es handelt sich übrigens meines Erachtens um ein Gerücht, daß GEM nur Resourcen von maximal 32KBytes verarbeiten kann. Vielmehr kann das Resource-Construction-Set 1.x nur Resource-Dateien dieser Größe erstellen. Schaut man in das RCS 2.x, so sagt einem der Menü-Punkt 'INFO...' gleich einen verbleibenden Arbeitsspeicher von 63000 Bytes voraus (warum das aber nicht die vollen 64KBytes abzüglich des Headers sind, weiß ich auch nicht). Übrigens sei darauf hingewiesen, daß beispielsweise Wercs zuläßt, Resourcen mit über 64KBytes zu 'erstellen' - aber wehe dem, der die Datei später wieder laden möchte...


```
000000  00 01 02 30 01 F2 02 0E 01 E0 01 DC 00 24 00 5C   .....$.\
000010  01 EE 02 90 00 04 00 01 00 01 00 01 00 01 00 01   .........
000020  00 01 02 94 45 69 6E 20 54 45 44 49 4E 46 4F 2D    TEDINFO-
000030  45 6C 65 6D 65 6E 64 00 00 00 4D 61 78 6F 6E 00   ...Maxon.
000040  75 6E 64 20 65 69 6E 20 53 74 72 69 6E 67 00 46    String.F
000050  72 65 65 2D 73 74 72 69 6E 67 00 00 00 00 00 00   ing......
000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000070  00 00 00 0C 00 18 30 00 00 00 00 1E 00 38 70 00   ......0......8p.
000080  00 00 00 1E 00 78 F0 00 00 00 00 0C 00 79 B1 E1   .....x.......y..
000090  8C 3F 3F 00 00 6B 23 39 9C E3 3D 00 00 EB 62 08   .??..k#9..=...b.
0000A0  F8 C3 39 80 00 CE 62 18 F0 C3 21 80 01 CC 66 30   ..9...b...!...f0
0000B0  70 C2 61 00 01 80 46 30 FC E6 C3 00 63 00 C7 F9   p.a...F0....c...
0000C0  DC 7C C2 00 F3 00 C0 00 00 00 00 00 F0 00 00 00   .|..............
0000D0  00 00 00 00 60 00 00 00 00 00 00 00 00 00 00 00   ....`...........
0000E0  00 00 00 0C 00 18 30 00 00 00 00 1E 00 3C 78 00   ......0......<x.
0000F0  00 00 00 3F 00 7E FC 00 00 00 00 7F 00 FF FD E1   ...?.~..... ....
000100  8C 3F 3F 7F 01 FF FF FB DE FF FF BF 01 FF FF FF   .?? ............
000110  FF FF FF DE 01 FF FF FF FF FF FF CC 03 FF FF FF   ................
000120  FF FF FF E0 03 FF FF FF FF FF FF E0 67 FF FF FD   ............g...
000130  FF FF FF C0 F7 FF FF FF FF FF FF E0 FF CF FF FF   ................
000140  FF FF FF E0 FF C3 FF FF FE FF E7 C0 FF 81 E7 F9   ..............ü..
000150  DC 7C C3 00 FB 00 C0 00 00 00 00 00 00 00 00 00   .|..............
000160  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000170  00 00 00 0C 00 18 30 00 00 00 00 1E 00 38 70 00   ......0......8p.
000180  00 00 00 1E 00 78 F0 00 00 00 00 0C 00 79 B1 E1   .....x.......y..
000190  8C 3F 3F 00 00 6B 23 39 9C E3 3D 00 00 EB 62 08   .??..k#9..=...b.
0001A0  F8 C3 39 80 00 CE 62 18 F0 C3 21 80 01 CC 66 30   ..9...b...!...f0
0001B0  70 C2 61 00 01 80 46 30 FC E6 C3 00 63 00 C7 F9   p.a...F0....c...
0001C0  DC 7C C2 00 F3 00 C0 00 00 00 00 00 F0 00 00 00   .|..............
0001D0  00 00 00 00 60 00 00 00 00 00 00 00 00 00 00 4F   ....`..........O
0001E0  00 00 01 5C 00 08 00 10 00 00 00 00 00 01 00 00   ...\............
0001F0  01 E0 00 00 00 24 00 00 00 38 00 00 00 39 00 03   .....$...8...9..
000200  00 06 00 02 11 80 00 00 FF FF 00 14 00 01 00 00   ................
000210  00 DC 00 00 00 5C 00 00 00 3A 10 31 00 00 00 00   .....\...:.1....
000220  00 00 00 00 00 40 00 10 00 11 00 18 00 1E 00 08   .....@..........
000230  FF FF 00 01 00 03 00 14 00 00 00 10 00 02 11 00   ................
000240  00 00 00 00 00 42 00 10 00 02 FF FF FF FF 00 16   .....B..........
000250  00 00 00 00 00 00 01 F2 00 02 00 02 00 13 00 01   ................
000260  00 03 FF FF FF FF 00 1F 00 00 00 00 00 00 02 0E   ................
000270  00 07 00 04 40 00 20 00 00 00 FF FF FF FF 00 1C   ....@. .........
000280  00 20 00 00 00 00 00 40 00 04 00 07 00 0E 00 01   . .....@........
000290  00 00 02 30 00 00 00 00 00 00 00 00 00 00 00 00   ...0............
```

Abbildung 1: Eine Resource-Datei in Zahlen

Blöckche' für Blöckche'...

Die Zeiger für die neun verschiedenen Blöcke innerhalb der Resource-Datei findet man im Anschluß an die Versionsnummer innerhalb des Resource-Headers. Der erste (relativ auf den Anfang bezogene) Zeiger zeigt auf die Objektstruktur. Aus der Tatache der Verzeigerung ergibt sich schon, daß der Block nicht an einer festgelegten Stelle innerhalb der Datei liegen muß. Dennoch habe ich festgestellt, daß die unterschiedlichen Resource-Construction-Sets die gleiche Reihenfolge einhalten (garantieren kann ich das natürlich nicht). Hinter den neun Blockzeigern findet man im Resource-Header Variablen, die die Anzahl der einzelnen Elemente innerhalb der Blöcke angeben. Schauen wir also zunächst einmal nach, wieviel Baumstrukturen und Objekte wir in der Objektstruktur finden werden. Der elfte Eintrag sagt aus, daß es vier Objekte sind, während aus dem zwölften Eintrag ersichtlich ist, daß in der Resource-Datei nur ein Baum zu finden ist. Daraus ergibt sich, daß der Block für die Objektstruktur viermal der Länge eines Objektes, welches 24 Bytes hat, also 96 ($60) Bytes entsprechen muß. Die Adresse des Objektblocks finden wir an der zweiten Stelle im Resource-Header direkt hinter der Versionsnummer. In unserem Beispiel ist dies die Adresse $230. Addieren wir die $60, kommen wir auf die Adresse $290, an der demnach ein weiterer Block beginnen müßte. Dies ist auch der Fall, da wir diese Adresse in der zehnten Variablen finden. Dieses Element des Resource-Headers enthält den Zeiger auf die sogenannte Baumindex-Tabelle.

Auf der Suche nach dem Baum

Was aber ist eine Baumindex-Tabelle? Stellen Sie sich vor, die Datei enthielte mehrere Bäume, dann müßten Sie doch wissen, wo innerhalb der Objektstruktur die einzelnen Bäume anfingen, das heißt, welche Nummer das Vaterobjekt der verschiedenen Bäume hat. Jetzt können Sie zwar entgegenhalten, daß Sie diese Adresse doch durch rsrc_gaddr() erfahren können. Wo aber hat dann diese AES-Routine die Adresse her? Ganz einfach: In dieser Baumindex-Tabelle findet man zu jedem Baum (in unserem Beispiel ist es nunmal nur ein Baum) die Adresse, an der der Baum beginnt. Würde die Datei beispielsweise drei Bäume enthalten, würden Sie dies auch an einer 12 Bytes langen Baumadreßtabelle erkennen. Jetzt fragt sich natürlich der schlaue Mann oder die schlaue Frau, warum denn bitteschön HIER Zeiger von 4 Bytes Länge spendiert wurden. Das liegt wahrscheinlich daran, daß genau an dieser


Chemo - Soft
Nadorster Str. 81 * 2900 Oldenburg
☎ (0441) 82851 ✱ FAX 86019
BTX (0441) 82851
BEWÄHRTES ZU FAIREN PREISEN
SDOmerge/index je 45,-
SDO PreView 45,-
Script 188,-
STAD V1.3+ 169,-
ESPRIT 'das Spiel' 94,-
MegaPaint 439,-
MegaPaint Profess. 699,-
Grafstar 83,-
Arabesque 229,-
ThemaDat 4.0 220,-
Headline 3.0 89,-
ConVert 89,-
Tempus 2.0x 104,-
Diskus 1.x 133,-
ST Pascal plus 220,-
WERCS 90,-
FTL-Modula-2 249,-
D.R.A.W. 127,-
MORTIMER 77,-
EASYBASE 243,-
HARLEKIN 119,-
LDW Power Calc 215,-
Beckerpage 2.0 319,-
Calamus 1.09 691,-
Lavadraw 137,-
1ST Word 3.15 222,-
1ST Proport. 3.1 109,-
Outline Art 345,-
PKS Write 179,-
Vektor Fonteditor 179,-
DIDOT 185,-
Calamus Fonts ab 39,-
That's Write 290,-
dito 'Junior' 139,-
THAT'S PIXEL 139,-
MEGASCREEN 291,-
ST Math/Analog je 87,-
ST Chemplot 87,-
Top Secret 'Platte' 170,-
NeoDesk 2.05 78,-
Turbo ST 1.8 78,-
Adimens 3.x 320,-
Aditalk 3.x 320,-
Adiprog f. GFA 145,-
AdiExtra Etikette 135,-
GFA-EWS 3.0x 170,-
GFA-EWS 3.5 233,-
GFA GUP/Assembl. je 125,-
GFA Assembler 1.3 125,-
1ST LEKTOR 146,-
HyperCACHE+ 16MHz 549,-
PC-Speed/STF 419,-
AT-Speed a.A.
Vortex ATonce 439,-
Datajet 30 1279,-
Datajet R44 2369,-
Handyscanner 105mm 385,-
dito 128mm 479,-
Für Bastler:
Advantage ST 253,-
Seagate ST1096 960,-
Seagate ST 157N-1 750,-
Gehäuse + Netzteil 189,-
Thermoreglerplatine 39,-
DMA Kabel 0,9m 39,-
DMA Kabel 1,2m 45,-
Speichererweiterungen
STE auf 2MB 479,-
dito auf 4 MB 959,-
1040/Mega1 > 2.5 649,-
dito > 4 (steckb.) 1029,-
Wechselpl.medien 240,-
Chemie-Software
CHEMOTECH V1.2 139,-
dito. 'SPEZIAL' 199,-
KRISTALLOTECH 1.3 69,-
Chemie Art Disk je 15,-
Bio-Art Disk je 15,-
Chemgraf 75,-
CHEMPLOT 87,-
Wissenschaftl. Softw.:
ProSTatist 170,-
GFA Statistik 889,-
dito 'Student' 379,-
Riemann 229,-
WIR FÜHREN NOCH VIEL MEHR !!!
MUSIKMASTER V1.0 - die Medienverwaltung 79,-
GEM-gesteuert verwalten Sie LP's, CD's und MC's und sortieren die Daten nach Interpret oder Jahr - oder suchen Sie nach Interpret, Titel oder Jahr !
komfortabler Eingabe-Editor / Cassetten-Cover oder Dateien drucken
Erstellen einer Monogramm-Bibliothek - INFO anfordern !
Kegeler-Verwaltungsprogramm 69,-
Einzel-Disziplinen-Verarbeitung - dadurch hoch flexibel u. Einstellungsmögl.
Statistikfunktionen & Listenausdruck & Auswertung auf Mon. od. Drucker
Schützenverwaltungsprogramm 2.0 129,-
analoge Möglichkeiten wie beim Kegelerverw.programm
wesentlich erweiterte und überarbeitete Version :
es wurde eine komplette Mitgliederverwaltung implementiert
Listenausdruck: Jubiläums-, Adressen- u. Geburtstagslisten, Lastschriften
Ausdruck aller gezahlten Mitgliederbeiträge (Kalkulation)
Info anfordern ! Die Demo-Disk erhalten Sie für 10 DM
Versandkosten 6 DM [Nachnahme + 5 DM!]. Bei Vorauskasse [Scheck/Einzugserm.] erhalten Sie 3% Barzahlungsrabatt bei Software und 2% bei Hardware ! Preisliste gratis. Preise & Lieferzeit vorbehalten! Weitere Produkte auf Anfrage - sprechen Sie mit uns!

Stelle später nach einem rscr_load die absoluten Adressen eingetragen werden, die ja 4 Bytes benötigen. Jeder weitere Zugriff wird dann relativ zu diesen Adressen durchgeführt. Wenn wir jetzt den Teil von rsrc_gaddr nachbilden wollten, der die Baumadresse berechnet, ergäbe sich folgende Routine:

```
OBJECT rsrc_gtreeadr(rsc_adr,baum_nr)
long rsc_adr;
int baum_nr;
{
   RSHDR *rshd;
   OBJECT *tree_index;

   rshd= (RSHDR*) rsc_adr;
   tree_index = (OBJECT*)(rsc_adr+rshd->rsh_trindex);

   return(tree_index[baum_nr]);
}
```

Listing 2

Damit sind wir nun in der Lage, die Anfangsadressen der Bäume innerhalb der Resource-Datei zu ermitteln (natürlich nur, solange sie noch nicht durch rsrc_load in den Speicher geladen wurde!). Dies ist also eine der Arbeiten, die von rsrc_load durchgeführt werden. Übrigens gibt es noch zwei weitere Index-Tabellen, in denen Adressen zu finden sind. Dies gilt nämlich auch für die sogenannten Free-Images und Free-Strings, auf die ich noch weiter unten zu sprechen komme.

Objekttypisches

Da wir nun wissen, wo unsere Bäume beginnen, wollen wir uns diese Objekte auch anschauen. Dabei gehe ich davon aus, daß Sie mit Objektbäumen vertraut sind und wissen, wie die Verknüpfung der einzelnen Objekte über ob_next, ob_head und ob_tail funktioniert (sie finden die Objektstruktur der Übersicht wegen aber nochmal in Listing 3 abgebildet). Auf diese Elemente wie auch auf Objekttyp, -status und -Flags muß ich nicht weiter eingehen, da diese unverändert vorliegen. Wie sieht es aber beispielsweise mit dem Eintrag ob_spec aus? Dieser muß, abhängig von der Art des Objektes, bearbeitet werden, wobei es sich dabei im wesentlichen um das Umrechnen von relativen in absolute Zeiger dreht. Am besten gehen wir dazu die einzelnen Objekttypen durch, die sich aber ganz gut in Gruppen zusammenfassen lassen. Zunächst einmal eine gute Nachricht: G_BOX, G_IBOX und G_BOXCHAR müssen bezüglich des Eintrags ob_spec nicht verändert werden. Die einfachsten Objektarten bezüglich des Umrechnens von ob_spec sind G_TITLE, G_STRING und G_BUTTON. Hier muß nur zu der relativen Adresse, die in ob_spec zu finden ist, die absolute Anfangsadresse der Datei im Speicher addiert und an die Stelle von ob_spec geschrieben werden. Schauen wir uns beispielsweise in Abbildung 1 das Objekt mit der Nummer 4 an, das an der Adresse $278 beginnt und einen String (Objekttyp $1C) darstellt, so erkennen wir in ob_spec (Adresse $284) die Adresse $0000 0040. Schauen wir nun an Adresse $40 nach, so finden wir - voilà - den Text "und ein String".

Das Umrechnen von G_TEXT, G_BOXTEXT, G_FTEXT und G_FBOXTEXT ist schon etwas aufwendiger, da in diesem Fall ob_spec auf die Tedinfo-Struktur zeigt, die wiederum Zeiger enthält, die wir natürlich auch bearbeiten müssen. Zunächst einmal wird also die Adresse aus ob_spec herausgelesen, die auf Tedinfo zeigt. Dann rechnen wir die drei Zeiger te_ptext, te_ptmplt und te_pvalid in absolute Adressen um, indem wir wie oben zu der relativen Adresse die absolute Anfangsadresse unserer Resource-Datei im Speicher hinzuaddieren, und tragen sie dann an die entsprechenden Stellen in Tedinfo als absolute Adressen ein. Zu guter Letzt rechnen wir noch ob_spec um.

Ähnliches wie bei TEDINFO gilt eigentlich auch für die Strukturen von G_ICON und G_IMAGE. Hier werden zunächst die Zeiger in den Strukturen (ib_pmask, ib_pdata, ib_ptext und bi_pdata) und danach der ob_spec-Zeiger umgerechnet.

Freiheit für Bild und Text

Übrigens: Ist Ihnen aufgefallen, daß wir die meisten Zeiger unseres Resource-Headers (rsh_tedinfo, rsh_iconblk usw.) überhaupt nicht benötigen? Das liegt daran, daß diese Zeiger normalerweise nur für die rsrc_gaddr()-Routine des AES sowie für das Resource-Construction-Set interessant sind, denn die brauchen die Information, wo der gesamte Block innerhalb der Datei steht. Trotzdem sind noch zwei Zeiger übriggeblieben, über die ich noch kaum etwas gesagt habe, nämlich rsh_frimg und rsh_frstr. Das ist nämlich etwas ganz Trickreiches. Ist Ihnen nicht ab und zu einmal dieses Free-Icon im Resource-Construction-Set 2.x aufgefallen (bitte nicht mit dem Free-Icon des RCS 1.x verwechseln, was ab 2.x-Version PANEL heißt!)? Hier können Sie Bilder (Images) oder Strings ablegen, ohne daß diesen eine Objektstruktur zugewiesen wird. Das heißt, diese Bilder und Texte liegen ohne Bezug zu einer Objektstruktur in der Resource-Datei. Nachtigall, ick hör dir trappsen: Was ist denn dann der Unterschied zwischen einer Alertbox und einem Free-String, bitteschön? Eigentlich keiner! Der Unterschied ist einfach der, daß eine Alertbox einen String in einem bestimmten Format erwartet. Erkennt also das RCS einen Free-String, der das Format "[Nummer][Text][Knopf1|..]" hat, so interpretiert es diesen als Alertbox! Sie können es ja spaßeshalber einmal ausprobieren. Nehmen Sie einen Free-'Baum', legen Sie einen Free-String hinein und geben den Text "[1][Wie geht's?][Gut|Schlecht]" ein. Danach müssen Sie allerdings erst die Datei abspeichern, aber gleich danach werden Sie ein Free-Icon weniger und eine Alertbox mehr haben!

Normalerweise kann man diese Free-Strings sehr gut dafür gebrauchen, jeden Text, den man im Programm verwendet, als einen solchen String abzulegen. Dadurch kann das Programm sehr einfach übersetzt werden, ohne daß man neu compilieren müßte (natürlich muß man auf Längenveränderungen der Texte eventuell vorbereitet sein). Free-Images sind insofern praktisch, als man in der Resource-Datei Bilder ablegen kann, die man später im Austausch gegen andere Bilder in die Objektstruktur einbaut. Der

```
typedef struct object{
   int ob_next;               /* Zeiger auf nächstes Objekt */
   int ob_head;               /* Anfang des Kindes */
   int ob_tail;               /* Ende des Kindes */
   unsigned int ob_type;      /* Objekttyp */
   unsigned int ob_flags;     /* Objekt-Flags */
   unsigned int ob_state;     /* Objektstatus */
   char *ob_spec;             /* Zeiger auf Zusatzinformationen */
   int  ob_x;                 /* X-Koordinate */
   int  ob_y;                 /* Y-Koordinate */
   int  ob_width;             /* Breite */
   int  ob_height;            /* Höhe */
} OBJECT;
```

Listing 3: Die Objektstruktur

große Vorteil der FREEs ist, daß sie keine Objekt- und Bitblk-Struktur pro Free verbrauchen (das sind 24 Bytes bei Texten und 24+14 Bytes bei Bildern)!

Zugriff auf die Freiheit

Nun stellt sich aber zugegebenermaßen die Frage, wie man dann an diese Free-Strings und -Images herankommt. Nichts leichter als das: Auch dafür gibt es zwei Tabellen, die wie die Baumindex-Tabelle organisiert sind, und daher je ein Feld von Langwortadressen darstellen (natürlich liegen diese Adressen nach dem Laden noch relativ vor und müssen erst in absolute Adressen umgerechnet werden). Die relativen Adressen der Tabellen bezogen auf den Dateianfang findet man, wie nicht anders zu erwarten war, in dem Header der Resource-Datei in den Elementen rsh_frimg und rsh_frstr. Übrigens liegen die Free-Strings in dem Block der anderen Strings und Texte (Zeiger rsh_string), und die Free-Images liegen vereint mit den anderen Bildern von Icons und Bitimages in dem Imagedata-Block (Zeiger rsh_imdata). Die Routine zum Ermitteln der Adresse eines Free-Strings ist relativ einfach und entpricht im Prinzip der rsrc_gtreeadr-Routine():

```
long rsrc_gfreeadr(rsc_adr,str_nr)
long rsc_adr;
int str_nr;
{
   RSHDR *rshd;
   long *free_index;

   rshd= (RSHDR*) rsc_adr;
   free_index = (long*)(rsc_adr+rshd->rsh_frstr);

   return(free_index[str_nr]);
}
```

Listing 4

Denken Sie daran, daß die Nummern der Alertboxen in der Definitionsdatei nicht den Index in das String-Feld darstellen, sondern daß es sich dabei um den Index in das Free-String-Feld handelt und Sie erst aus diesem Feld (indirekt) die Adresse auf den String der Alertbox erhalten. Die Adresse eines Free-Images errechnet man genauso wie in Listing 4 dargestellt, nur mit dem Unterschied, daß man nicht auf rsh_frstr, sondern auf rsh_frimg indiziert. Sollten Sie diese Images als Austauschbilder für Icons oder Bitimages verwenden wollen, müssen diese Bilder natürlich auch austauschfähig sein. Da sie ja keine Definitionsstruktur (BITBLK oder ICONBLK) haben, sollten sie in dem Format sein wie das Bild, das Sie austauschen, oder es ist an Ihnen, die Definitionsstruktur des auszutauschenden Bildes so zu manipulieren, daß es wieder stimmt.

Koordinieren will gelernt sein

Jetzt habe ich schon Unmengen an Zeilen damit verbraucht, mich mit Zeigern herumzuschlagen, wo es doch noch eine andere wichtige Sache zu erzählen gibt. Ist Ihnen aufgefallen, um welche Objekteinträge wir uns noch nicht gekümmert haben? Richtig, um die Koordinaten! Schaut man sich die Koordinaten (wobei ich hier auch Höhen- und Breitenangaben miteinbeziehe) in der RSC-Datei an, wird man sich zunächst einmal wundern, daß diese (bis auf Ausnahmen, siehe unten) relativ klein sind - besonders auffällig ist das bei der Höhe und Breite. Schauen Sie sich beispielsweise das oben schon erwähnte vierte Objekt an, das wir als String-Typ identifiziert hatten. Der letzte Eintrag des Objektes ist die Höhe, die, o Wunder, mit EINS angegeben wurde. Kennen Sie ein String-Objekt mit der Höhe eines Pixels? Ich nicht! Demnach kann die Angabe nicht in Pixeln gemeint sein, sondern ...in Buchstabeneinheiten. Das bedeutet, daß alle X-Koordinaten und die Breite nach dem Laden der Resource-Datei zunächst mit der Breite der aktuellen Schrift und die Y-Koordinaten sowie die Höhe mit der Höhe der aktuellen Schrift multipliziert werden müssen. Das bedeutet natürlich, daß beispielsweise eine quadratische Dialogbox, die in einer Auflösung erstellt wurde, die eine Schrift mit den Ausmaßen 8x16 hat (hochauflösende Grafik), alles andere als quadratisch in einer Auflösung sein wird, die eine Schriftgröße von 8x8 hat. Beachten Sie dabei auch, daß Sie nicht nur die Koordinaten in den Objektstrukturen, sondern auch die in den anderen Strukturen wie ICONBLK, BITBLK und TEDINFO umrechnen!! Jetzt wissen Sie auch, warum die Resource-Construction-Sets die Objekte immer auf Buchstabenkoordinaten 'einschnappen' lassen.

Eingeschnappte Objekte...

Wie immer gibt es natürlich eine Ausnahme: Wenn Sie sich an das FREE-Icon des RCS 1.x oder an das PANEL-Icon des RCS 2.x erinnern und schon einmal damit gearbeitet haben, wissen Sie vielleicht noch, daß die Objekte dort nicht auf Buchstabenkoordinaten einrasten. Das wirft zwei Fragen auf: Erstens wird man sich fragen, wie diese Koordinaten kodiert werden, da in der Resource-Datei nicht angegeben werden kann, um welche Baumart es sich handelt; und zweitens stellt sich die Frage, was man von dem Einschnappen oder dem Feinpositionieren hat. Beantworten wir erst die zweite Frage: Das Feinpositionieren hat natürlich den Vorteil, daß Sie Objekte näher und anders aneinanderrücken können, als dies der Fall beim Einschnappen der Objekte ist. Der Nachteil ist, daß Sie abhängig von der Bildschirmauflösung sind, denn ein Pixel nach links gerückt in einer Grafik, deren Buchstaben 32 Pixel breit sind, hat andere Auswirkungen als in einer Grafik, deren Buchstaben nur 8 Pixel breit sind. Ergo, man muß, wenn man diese Feinpositionierung und damit eine höhere Objektdichte erreichen möchte, für jede Grafikauflösung eine eigene Resource-Datei erstellen. Das ist ohne Probleme möglich, bedeutet aber einen recht hohen Aufwand. In einer 'eingeschnappten' Objektart sind die Abstände der Objekte immer relativ zu unterschiedlichen Auflösungen gesehen gleich.

Nun aber zu der Frage, wie die Koordinaten der Feinpositionierung abgespeichert werden. Ganz einfach: Die GEM-Autoren gingen davon aus, daß die Grafikauflösungen nicht so groß werden, daß darauf mehr als 256 Buchstaben nebenoder untereinander stehen werden (was zugegebenermaßen auch relativ wahrscheinlich ist). Das bedeutet aber, daß man nur das untere Byte der Koordinaten-Variablen benötigt. Das obere Byte wird zur Feinpositionierung verwendet, wobei es die zusätzliche Verschiebung der Koordinate nach rechts oder nach unten in Pixeln angibt. Das Errechnen der Koordinaten ergibt sich also beispielsweise zu

```
x= (ob_x & 0xff)*gl_wchar + (ob_x >>8) & 0xff
```

und

```
y= (ob_y & 0xff)*gl_hchar + (ob_y >>8) & 0xff
```

wobei Sie sicherlich noch wissen, daß man gl_hchar und gl_wchar, die die Höhe und Breite der verwendeten Schrift angeben, über das VDI erfahren kann. Noch einmal sei der Hinweis angebracht, daß das Erstellen von feinpositionierten Dialogboxen mehr Arbeit erfordert und zu Grafik-Inkompatibilitäten führen wird.

Aussichten

Sicherlich bietet das Resource-Format einige interessante Eigenschaften, die man auch recht gut nutzen kann, wie bei-





NEU
Version 1.5

Einschreiber

That's Write - Junior

Textverarbeitung für Einsteiger

Neben dem Können der **Junior**-Version ist besonders der günstige Preis eine Entscheidung für alle, die **leicht** und **richtig** einsteigen wollen.

- **Flexibel,** d.h. einfacher Umgang mit einem komfortablen, **preisgünstigen** Textverarbeitungsprogramm. Jederzeit Aufstieg in die Profi-Version möglich
- **Graphikeinbindung**
- **Seitenlayout**
- **Absatzlayout**
- **Silbentrennung** nach deutschen Regeln, erweiterbar durch Eingabe eigener Wörter in unbegrenzter Höhe
- Bis zu 10 verschiedene **Fonts** (Schriftarten) in einem Text möglich
- Lieferung mit **10 Fonts (10 versch. Schriften gleich dabei)**
- **Ausdruck im Graphik- und Text-Modus** (auch gemischt innerhalb einer Zeile)
- **Blockfunktionen**
- **Suchen und ersetzen**
- **Tabulatoren**
- **ASCII einlesen und speichern**
- **Bilddarstellung entspricht dem Ausdruck** (Keine Eingabe von Steuerzeichen nötig)
- **Leicht verständliches deutsches Handbuch**

Preis: DM **148,-**

Aufstieg

Jederzeit **Aufstieg in die Profi-Version** möglich (Original-Diskette der Junior-Version an Heim-Verlag einsenden und Profi-Version bestellen)

Preis: DM **246,-**

Alle genannten Preise sind unverbindlich empfohlene Verkaufspreise.

Kundenbetreuung durch
Compo Software GmbH

Informationen
und
Hotline

Telefon: 0 65 51 - 62 66

Vielschreiber

That's Write

Die professionelle Textverarbeitung

Endlich! Ein Textverarbeitungsprogramm, das neben seinem Können auch anwenderfreundlich ist.

That's Write ist für alle gängigen Anwendungen einer Textverarbeitung geeignet. Dank der einfachen Bedienung per Maus ist es für den Einsteiger leicht und schnell zu erlernen. Dem viel schreibenden Anwender bietet es die Bedienung über die Tastatur oder die Definition eigener Funktionen über Makros.

Komplettpaket mit großem Funktionsumfang:
• eigener Fonteditor • internationale mehrfach frei belegbare Tastatur • alle Tasten mit Floskeln und Makros belegbar • zuverlässige eingebaute Silbentrennung • umschaltbare Trennregeln für verschiedene Sprachen • eingebaute Rechtschreibkorrektur • das Hauptwörterbuch kann gewechselt werden (Option Englisch, Niederländisch, weitere in Vorbereitung) • eingebaute Serienbrieffunktion mit Schnittstelle zu Datenbanken und Adressverwaltungen • integrierte Schnittstelle zu Accessories, Datenübernahme aus Adressverwaltung • autom. Erstellung eines Stichwortverzeichnises • autom. Erstellen eines Inhaltsverzeichnises • autom. Fußnotenverwaltung • autom. Endnotenverwaltung • Umbenennen, kopieren, löschen von Dateien vom Programm aus • Snapshot-Accessory erlaubt aus anderen Programmen Bilder vom Monitor abzugreifen und diese in That's Write zu verwenden • einfache Installation • Gliederungsfunktion

Optimale Druckerausnutzung:
• 9-Nadel-Drucker (alle Auflösungen) • 24-Nadel-Drucker (alle Auflösungen, auch 360x360 dpi) • Laserdrucker (alle Auflösungen) • Atari-Laserdrucker SLM804 über DMA
• Nur Textdruck bei Typenraddruckern (auch Proportionalschrift in Blocksatz!), verschiedene Typenräder anpaßbar

Ausdruck von Text- und Graphiktextmodus in einer Zeile:
• optimaler Randausgleich • unterstützt interne Schriften des Druckers, wie Proportional, Pica, Elite, schmal, breit, doppelt hoch • unterstützt Grafikfonts freier Größe • unterstützt beliebig viele Download-Fonts in einem Text • unterstützt druckerinterne Vektorfonts (Kyocera u.a.)

Durchdachte Bedienung:
Erstmalig ein übersichtliches und schnelles Bearbeiten von selbst langen Texten. z.B.: „Text zuklappen" versteckt in Sekundenbruchteilen den normalen Text – und zeigt nur noch die Kapitelüberschriften. Dort geht man schnell in das gewünschte Kapitel und „klappt" wieder auf. Noch schneller geht es wohl wirklich nicht mehr.

Einzigartiges Druckkonzept:
That's Write druckt Grafiktextmodus und Textmodus gleichzeitig! Wer bietet mehr?

Optimierte Seiteneinteilung beseitigt drei klassische Probleme automatisch:
• Umbruch zwischen Überschrift und Text wird automatisch verhindert
• Leerraum nach Seitenumbruch wird nicht ausgedruckt, d.h. gleichmäßiger oberer Rand
• Einzelne Zeilen eines Absatzes alleine auf einer Seite werden automatisch verhindert.

Bildschirmanzeige entspricht dem Ausdruck (WYSIWYG)

Einfachste Bedienung per Maus, Tastatur oder Makros

Flexible Textgestaltung durch Absatz- und Seitenlayouts

Weitere Vorzüge:
• Unterstützt selbständig den Blitter • läuft unter allen bekannten TOS-Versionen und unter GEM 2.2 • unterstützt Farb- und Monochrombildschirm sowie Großbildschirme • korrekte Behandlung von getrennten Wörtern bei Suchen, Ersetzen und Rechtschreibekorrektur • zwei Texte gleichzeitig bearbeitbar, schnelles Wechseln zwischen den Texten • autom. Sichern während des Schreibens in einem einstellbaren Zeitintervall • Einfügen von Kurz-Datum, Lang-Datum, Seitennummer, Folgeseite per Tastendruck • Textstatistik – Anzahl der Wörter, Zeilen, Seiten, Bilder • Zählfunktion für Wörter (und/oder Wort-Teile) • Ausschneiden/Einfügen von Blöcken mit 4 unabhängigen Puffern • Einfüge- und Überschreibmodus auch bei Proportionalschrift • flexibles ASCII-Laden/Sichern zeilen/absatzweise • Textübernahme von 1st Word/Wordplus. Vorhandene Texte werden mit Fußnoten und allen Attributen wie fett, kursiv, hoch/tief, unterstrichen eingelesen • 10 Marken zum schnellen Anspringen von Textpassagen, die mit dem Text abgespeichert werden • Hilfsfunktionen wie „Wort klein schreiben", „Wort groß schreiben", „1. Buchstabe groß/Rest klein" zur schnellen Korrektur von Tippfehlern. • komfortables Bewegen im Text: zeichenweise, wortweise, zeilenweise, absatzweise, Bildschirm-seitenweise Textseitenweise (zur Kontrolle des Seitenumbruchs), Text Anfang/Ende, Block Anfang/Ende, Seitenanfang
• Druckausgabe auf Druckerport (schnelle Direktansteuerung), Druckerport über Bios, RS232 oder Datei – Bilder können beim Drucken auch weggelassen werden (Probeausdrucke, usw.) • So lassen sich zuerst „gerade" und später „ungerade" Seiten ausdrucken, um Papier komfortabel beidseitig zu bedrukken, ohne jedes Blatt einzeln wenden zu müssen
• Hotline für registrierte Anwender

Preis: DM **348,-**

Änderungen von That's Write 1.3 auf Version 1.5:

1. Rechtschreibkorrektur nach Langenscheidt:
   a) Das deutsche Wörterbuch kennt ca. 3 Millionen Wörter
   b) Trennung nach Wörterbuch möglich, daher erstmalig auch korrekte Trennung von zusammengesetzten Wörtern: „Gas-tank", „Hals-tuch"
   c) Fehler bei Groß-/Kleinschreibung werden, wenn eindeutig, automatisch erkannt und behoben!
   d) Sinnvolle Korrekturvorschläge, so wird z.B. bei „Füsiker" nur das Wort „Physiker" vorgeschlagen.
   e) Einfaches Erweitern und Bearbeiten von Zusatzwörterbüchern und Trennungen.
   f) Zusätzliche Wörterbücher erhältlich: Französisch, Englisch, Amerikanisch, Spanisch, Schwedisch, Schweizer-Deutsch
2. Textverkettung – jetzt kann per Knopfdruck in den folgenden oder den vorigen Text gewechselt werden. Seitennummern und Fußnotennummern werden automatisch übernommen
3. Fonts löschen, Anschlagzählung
4. Paßwortschutz verhindert unberechtigtes Lesen von neugierigen, naseweisen Wesen
5. Textinfo speichert zum Text: Autor, Stichworte, Erstellungsdatum
6. Doppelseitiges Drucken, da linke und rechte Kopftexte (Heftrand kann berücksichtigt werden)

• That's Write Junior
• That's Write Profi
jeweils einschließlich
Signum* Font-Konverter

Weitere
That's Write Font-Disketten
lieferbar

spielsweise die Free-Objekte und auch die Feinpositionierung der Objekte. Leider waren die GEM-Designer etwas knauserig, was die Bytes in dem Resource-Header betrifft. Dadurch, daß hier nur zwei Bytes pro Zeiger spendiert wurden, lassen sich nun mal nur Zahlen bis 64K kodieren. Leider betrifft das natürlich auch den letzten Eintrag des Resource-Headers. Eine Möglichkeit möchte ich noch aufzeigen. Durch die Tatsache, daß die Datenblöcke in beliebigen Reihenfolgen in der Datei liegen dürfen, kann man sich auch Platz für eventuelle andere Informationen schaffen (wie wäre es etwa mit einer kodierten Seriennummer in einer Resource-Datei...), allerdings ist mir selbst noch keine Applikation bekannt, die so etwas ausnutzt.

Namensgebung

Sicherlich wird jedem von Ihnen, der schon einmal mit einem RCS gearbeitet hat, aufgefallen sein, daß dieses Programm nicht nur die RSC-Datei abspeichert, sondern netterweise auch noch zwei (eventuell auch mehr) andere Dateien. Zum einen handelt es sich dabei um die sogenannte Header-Datei (nicht zu verwechseln mit unserem Resource-Header). In dieser Header-Datei, die als C-Ausgabe die Extension '.H' hat, werden Namen bestimmten Konstanten zugeordnet, die man später zur einfacheren Handhabung in dem eigenen Programm verwenden kann. Möchte man z.B. die Adresse einer Baumstruktur erfahren, muß man der Routine rsrc_gaddr() die Nummer des Baumes übergeben. Viel einfacher funktioniert es aber, wenn man im RCS diesem Baum einen Namen gibt. In der Header-Datei steht dann dieser Name als Konstante (mit der entsprechenden Zahl) definiert. Anstatt nun die Nummer des Baumes, die sich ja auch eventuell verändern kann, rsrc_gaddr() mitzuteilen, übergibt man von nun an den Namen als Konstante. Verändert sich durch Eingriffe im RCS die Nummer des Baumes, wird beim Abspeichern der Resource-Datei auch die Header-Datei geändert, die beim Neucompilieren des Programms miteingelesen wird - und schon stimmt der Zugriff auf rsrc_gaddr() wieder. Natürlich können auch Namen für Objekte vergeben werden, was im allgemeinen sogar noch wichtiger ist als bei Bäumen, da sich die Objektnummern recht häufig ändern. Diese Header-Datei wird von den meisten RCS inzwischen schon für die unterschiedlichsten Programmiersprachen ausgegeben, so daß Konstanten nicht nur für C-Programme bereitstehen.

DEF- und RSD-Format von RCS 1.x und Kuma-Resource 2

```
Offset   Länge   Beschreibung
  0        2     Anzahl der Datensätze in der Datei
                 (nur im ersten Datensatz ungleich 0)
  2        2     Header-Ausgabe: 1=C, 2=PASCAL, 4=MODULA2,8=FORTRAN77
                 (nur im ersten Datensatz ungleich 0)
  4        1     Bei Objekten: Index des zugehörigen Baums
  5        1     Index des Baumes beziehungsweise Objektes
  6        1     1 wenn Objekt, 0 wenn Baum
  7        1     Typ-Code (siehe weiter unten)
  8        8     Name des Objektes/Baums mit Nullen aufgefüllt

Typ-Codes: 0 Objekt im Baum oder Menü
           1 alter Free-Typ (im RCS 2.x PANEL)
             wird in KResource2 nicht mehr verwendet!
           2 Menübaum
           3 Formularbaum
           4 Alertbox
           5 Free-String
           6 Free-Image
```

DFN-Format

```
Das DFN-Format entspricht im wesentlichen dem DEF-Format. Der Unterschied
liegt darin, daß die Hi- und Lo-Bytes der Einträge vertauscht werden (es
lebe INTELs Prozessorreihe!) und das Byte 3 und 4 entfällt. Diese Datei
ist kompatibel zum RCS auf MS-DOSe-Rechnern.
```

HRD-Format

```
Das Wercs-Format ist in einen Anfangsrecord und in der Länge sich ändernde
Folgerecords aufgeteilt. Beachten Sie, daß die Typ-Definition eine andere
im Gegensatz zu den anderen Definitiondateien ist:

Der Kopf der HRD-Datei:

Offset   Länge   Beschreibung

  0        2     Version: Momentan 1
  2        1     Auto-Namen: 1 wenn Wercs automatisch Namen vergeben soll.
  3        1     Sprach-Flag: 1=C, 2=PASCAL, 4=Modula2
                              8=FORTRAN, 16=Assembler, 32=BASIC
  4        1     Auto-Pos:    0 Auto-Pos ausgeschaltet
                              1 Halb-Pos eingeschaltet
                              2 Auto-Pos eingeschaltet
  5        1     Buchstaben:  0 Groß/Klein
                              1 GroP
                              2 Klein
  6        1     Auto-Größe   0 Auto-Größe aus
                              1 Auto-Größe ein
  7        1     reserviert

und das Folge-Record:

Offset   Länge   Beschreibung

  0        1     TYP: siehe unten
  1        1     reserviert
  2        2     Baum-Index, das heißt Nummer des Baums
  4        2     Objekt-Index
  6-    variiert Name des Objektes mit 0 abgeschlossen

Typ-Codes: 0 Dialog
           1 Menü
           2 Alarmbox
           3 Freier Text - Free-String
           4 Freier Bitblock - Free-Image
           5 Objekt
           6 Dateiende
           7 Wenn Record für Präfix steht (?)
```

Listing 5: Die Definitionsformate der Resource-Construction-Sets

Per Definition anders

Die dritte Datei ist die sogenannte Definitionsdatei, die nur für das Resource-Construction-Set von Bedeutung ist. Hier merkt sich das RCS im wesentlichen die Namen der Objekte, da es die Header-Datei nicht interpretieren kann, sowie die Art des Baumes, sprich das Icon, daß Sie als Baum-Typ ausgewählt haben. Ich kenne inzwischen vier Resource-Construction-Sets (wenn man mal das nicht mehr so gebräuchliche Megamax-RCS wegläßt), die die fantastische Eigenschaft besitzen, drei unterschiedliche Definitionsdateien zu haben. Das RCS 1.x besitzt die sogenannte 'DEF'-Datei, die mit der 'RSD'-Datei des Kuma Resource identisch ist, aber umbenannt wurde, da die Extension DEF schon für Include-

Dateien von Modula verwendet werden. Das RCS 2.x verwendet die 'DFN'-Datei, die sich dadurch von der 'DEF'-Datei des Vorgängers unterscheidet, daß innerhalb Datei die Zahlenkodierung geändert wurde (siehe Listing 6). Zwar liefert ATARI beim RCS 2.x ein Programm (DEF2DFN), mit dem man 'DEF'-Dateien in 'DFN'-Dateien umwandeln kann, aber umgekehrt ist dies nicht möglich. Einmal RCS 2.x, immer RCS 2.x. Das Format der WERCS-HRD-Datei ist gleich ganz anders, und ein Konvertieren ist nur in Richtung 'HRD'-Datei möglich und nicht wieder zurück. Alle Programmierer sind wahrscheinlich der Meinung gewesen, daß ihr Programm das einzig wahre und eine Rekonvertierung nicht wichtig sei. Deshalb habe ich mir die Mühe gemacht und alle Formate einmal zusammengestellt. Sie finden Sie in Listing 5. In Listing 6 enthält ein Programm, mit dem Sie alle Formate ineinander überführen können. Natürlich kann es dabei passieren, daß Sie Informationen verlieren, oder daß umgekehrt von dem Programm Informationen hinzugefügt werden müssen. Beim Hinzufügen (beispielsweise bei der Umsetzung ins HRD-Format) habe ich die Optionen verwendet, die mir am gebräuchlichsten erschienen. Für ProgrammiererInnen, die mit mehreren RCS arbeiten, wird dieses Utility sicherlich von großem Nutzen sein. Eine Bitte noch: Die Routinen sind aufgrund der bei den Programmen mitgelieferten Informationen über die Datenformate erstellt worden und könnten daher eventuell nicht ganz vollständig in der Konvertierung sein. Ich habe zwar selbst noch keine Probleme dabei gehabt, dennoch sehen die Konvertierungen der mitgelieferten Programme teilweise geringfügig unterschiedlich aus. Ich vermute, daß dort unsauber gearbeitet wurde. Vorsicht ist die Mutter der Porzellankiste. Wie gesagt, hatte ich bisher noch keine Probleme damit, sollten Sie aber relevante Unterschiede erkennen, bitte ich Sie, mir diese mitzuteilen.

MAIN
Resource-Definitions-Datei
Konverter
DEF2DFN
RCS 1.X -> RCS 2.X (->DFN)
DFN2DEF
RCS 2.x -> RCS 1.X (->DEF)
DEF2HRD
RCS 1.x -> Wercs (->HRD)
HRD2DEF
Wercs -> RCS 1.X (->DEF)
ABBRUCH
Abbruch

So muß die Dialogbox für den Konverter erstellt werden.

Zusammenfassend

Ich hoffe, daß ich mit diesem Artikel einen Einblick in die Untiefen der Resourcen und deren Konstruktionshilfen geben konnte. Durch die von mir gelieferten Informationen müßte Ihnen eigentlich klargeworden sein, welche Arbeiten die rsrc_load()-Routine durchführt und wie man sie für eigene Zwecke abgewandelt nachbilden kann. Viel Spaß dabei wünscht Ihnen

SH

```
 1:  /**********************************************/
 2:  /*                                            */
 3:  /* Resource-Definitionsdatei-Konverter        */
 4:  /* Version 1.0  Datum: 15.5.1990              */
 5:  /* Stefan Höhn                                */
 6:  /* konv: DEF->DFN, DFN->DEF, DEF->HRD,HRD->DEF */
 7:  /*       RSD entspricht DEF !                 */
 8:  /*                                            */
 9:  /**********************************************/
10:
11:  #include <obdefs.h>
12:  #include <gemdefs.h>
13:  #include <osbind.h>
14:  #include "defconv.h"
15:
16:  #define NONAME_GIVEN "[1][Sie haben keinen|Namen
                          angegeben!][Abbruch]"
17:  #define FILE_ERROR1  "[1][Datei gibt es nicht
                          !!!][Abbruch]"
18:  #define FILE_ERROR2  "[1][Datei gibt es schon
                          !!!][Abbruch]"
19:
20:  typedef struct hrd_deader  /* Kopf HRD-Datei */
21:  {
22:      int  version;   /* Versionsnr: momentan 1 */
23:      char auto_name; /* 0=aus/1=ein */
24:      char language;  /* 1=C,2=Pascal,4=Modula2,
                            8=Fortran,
25:                         16=Assembler,32=BASIC */
26:      char auto_pos;  /* 0=aus, 1=Halb-Pos. ein,
                            2=Voll-Pos. ein */
27:      char character; /* 0=GroP/Klein, 1=GroP,
                            2=Klein */
28:      char auto_size; /* 0=aus, 1=ein */
29:      char reserved;
30:  }HRD_HEADER;
31:
32:  char hrd_typ[]={5,0,1,0,2,3,4};   /* Umkodierung
                            RCS1.x- in Wercs-Typ */
33:  char def_typ[]={3,2,4,5,6,0};     /* Umkodierung
                            Wercs- in RCS1.x-Typ */
34:
35:  OBJECT *tree;
36:
37:  main()
38:  {
39:     int exit_obj, proc_ret, x,y,w,h, ret;
40:     char from_name[64], to_name[64], pname[64],
             fname[64], *srch="*.*";
41:     int  from_fd,       to_fd;
42:     GRECT box;
43:
44:     appl_init();              /* bei AES anmelden */
45:     graf_mouse(ARROW,0L);     /* Pfeil als Maus */
46:
47:     fname[0]=pname[0]=0;      /* Namen auf Null */
48:
49:     rsrc_load("defconv.rsc");
50:     rsrc_gaddr(0,MAIN,&tree);
51:     form_center(tree,&x,&y,&w,&h);  /* Box
                                     zentrieren */
52:     form_dial(0,0,0,639,200);       /* Der
                             Einfachheit halber. */
53:          /* Besser Koordinaten von VDI holen ! */
54:     form_dial(1,0,0,320,200,x,y,w,h); /* Box
                                         öffnen */
55:     objc_draw (tree,0, MAX_DEPTH,0, 0,639, 399);
                               /* Box zeichnen */
56:     exit_obj=form_do(tree,0);     /* bearbeiten */
57:
58:     if (exit_obj!=ABBRUCH)          /* Solange
                     bis Abbruch gedrückt wurde */
59:     {
60:        form_alert(0,"[1][Bitte wählen Sie
                     erst|die Quell- und dann| die
                     Zieldatei!][OK]");
61:        ret=do_file(pname,fname,srch,from_name);
                       /* Dateinamen einlesen */
```

→

```
 62:         if (ret && from_name[0])
                             /* wenn OK und Name!='' */
 63:         {
 64:            fname[0]=0;            /* Namen löschen */
 65:            ret=do_file(pname,fname,srch,to_name);
                               /* nächsten Namen holen */
 66:            if (!ret || !to_name[0])
                        /* Falls Abbruch oder kein Name */
 67:            {
 68:               form_alert(0,NONAME_GIVEN);
                                  /* Meldung ausgeben */
 69:               exit(-1);
 70:            }
 71:         }
 72:         else              /* Abbruch oder kein Name */
 73:         {
 74:            form_alert(0,NONAME_GIVEN);
 75:            exit(-1);
 76:         }
 77:
 78:         switch(exit_obj) /* Welche Konvertierung */
 79:         {
 80:
 81:            case DEF2DFN:
                 proc_ret=def2dfn(from_name, to_name);
 82:                         break;
 83:
 84:            case DFN2DEF:
                 proc_ret=dfn2def(from_name, to_name);
 85:                         break;
 86:
 87:            case DEF2HRD:
                 proc_ret=def2hrd(from_name, to_name);
 88:                         break;
 89:
 90:            case HRD2DEF:
                 proc_ret=hrd2def(from_name, to_name);
 91:                         break;
 92:         }
 93:      }
 94:      form_dial(2,0,0,320,200,x,y,w,h);    /* Box
                                            schliePen */
 95:      form_dial(3,0,0,639,399);
                     /* Bildschirmbereich freigeben */
 96:      tree[exit_obj].ob_state&=~SELECTED;  /* Bit
                                       zurücksetzen */
 97:
 98:      appl_exit();
 99:
100:      return(proc_ret);
101:   }
102:
103:   int def2dfn(from, to)    /* RCS 1.x/RSD -> RCS
                                  2.x konvertieren */
104:   char *from, *to;
105:   {
106:      int fd1, fd2, i;
107:      long flen;
108:      char byte1, byte2, symbol[8];
109:
110:      fd1=Fopen(from,0);            /* Datei öffnen */
111:      if (fd1<0)            /* ist nicht zu öffnen */
112:      {
113:         form_alert(0,FILE_ERROR1);        /* Meldung
                                           ausgeben */
114:         return(-2);
115:      }
116:      fd2=Fopen(to,0);      /* Ausgabedatei öffnen */
117:      if (fd2>0)              /* existiert bereits */
118:      {
119:         Fclose(fd1);        /* dat janze abbrechen */
120:         Fclose(fd2);
121:         form_alert(0,FILE_ERROR2);
122:         return(-3);
123:      }
124:      fd2=Fcreate(to,0);           /* Datei erzeugen */
125:      flen= Fseek(0L,fd1,2);   /* Länge ermitteln */
126:      Fseek(0L,fd1,0);     /* an den Anfang setzen */
127:      i=flen;
128:      while(i>0)
129:      {
130:         Fread(fd1,1L,&byte1);    /* 2 Bytes lesen */
131:         Fread(fd1,1L,&byte2);
132:         Fwrite(fd2,1L,&byte2);     /* vertauscht
                                            schreiben */
133:         Fwrite(fd2,1L,&byte1);
134:
135:         Fread(fd1,1L,&byte1);  /* 2 B. übergehen */
136:         Fread(fd1,1L,&byte2);
137:
138:         Fread(fd1,1L,&byte1);    /* 2 Bytes lesen */
139:         Fread(fd1,1L,&byte2);
140:         Fwrite(fd2,1L,&byte2);     /* vertauscht
                                            schreiben */
141:         Fwrite(fd2,1L,&byte1);
142:
143:         Fread(fd1,1L,&byte1);    /* 2 Bytes lesen */
144:         Fread(fd1,1L,&byte2);
145:         Fwrite(fd2,1L,&byte2);     /* vertauscht
                                            schreiben */
146:         Fwrite(fd2,1L,&byte1);
147:
148:         Fread(fd1,8L,symbol);
149:         Fwrite(fd2,8L,symbol);
150:         i-=16;
151:      }
152:      Fclose(fd1);           /* Dateien schliePen */
153:      Fclose(fd2);
154:      return(0);
155:   }
156:
157:   int dfn2def(from, to)  /* RCS 2.x -> RCS 1.x/RSD
                                   konvertieren */
158:   char *from, *to;
159:   {
160:      int fd1, fd2, i, null=0;
161:      long flen;
162:      char byte1, byte2, symbol[8];
163:
164:      fd1=Fopen(from,0);  /* Datei Lesen öffnen */
165:      if (fd1<0)          /* ist nicht zu öffnen */
166:      {                   /* Abbruch */
167:         form_alert(0,FILE_ERROR1);
168:         return(-2);
169:      }
170:      fd2=Fopen(to,0);    /* Datei  öffnen */
171:      if (fd2>0)          /* Datei gibt es schon */
172:      {                   /* Abbruch */
173:         Fclose(fd1);
174:         Fclose(fd2);
175:         form_alert(0,FILE_ERROR2);
176:         return(-3);
177:      }
178:      fd2=Fcreate(to,0);
179:      flen= Fseek(0L,fd1,2);  /* Länge ermitteln */
180:      Fseek(0L,fd1,0);         /* an den Anfangs
                                      setzen */
181:      i=flen;
182:      while(i>0)
183:      {
184:         Fread(fd1,1L,&byte1);    /* 2 Bytes lesen */
185:         Fread(fd1,1L,&byte2);
186:         Fwrite(fd2,1L,&byte2);     /* vertauscht
                                            schreiben */
187:         Fwrite(fd2,1L,&byte1);
188:
189:         Fwrite(fd2,2L,&null);
190:
191:         Fread(fd1,1L,&byte1);    /* 2 Bytes lesen */
192:         Fread(fd1,1L,&byte2);
193:         Fwrite(fd2,1L,&byte2);     /* vertauscht
                                            schreiben */
194:         Fwrite(fd2,1L,&byte1);
195:
196:         Fread(fd1,1L,&byte1);    /* 2 Bytes lesen */
197:         Fread(fd1,1L,&byte2);
198:         Fwrite(fd2,1L,&byte2);     /* vertauscht
                                            schreiben */
199:         Fwrite(fd2,1L,&byte1);
200:
201:         Fread(fd1,8L,symbol);
202:         Fwrite(fd2,8L,symbol);
203:         i-=14;
204:      }
205:      Fclose(fd1);
206:      Fclose(fd2);
207:      return(0);
208:   }
209:
210:   int def2hrd(from, to)  /* RCS 1.x/RSD -> Wercs
                                   konvertieren */
211:   char *from, *to;
```

→

DIGITAL DATA DEICKE, NORDRING 9, 3000 HANNOVER 1, TEL: 0511/6370 -54, -55

Preiswert speichern: HD-Floppy für alle ! (16 MByte für DM 29,- (=10 Stk. HD-Disketten))

NEU: ddd Farb-Multiscan für ATARI; anschlußfertig !

ATARI STE

Die Alternative für die Preisbewußten: Kompaktgerät mit 1, 2 oder 4 MByte RAM, 720 KByte Laufwerk, Betriebssystem TOS 1.6, D/A-Wandler zur Stereo-Sounderzeugung, 4096 Farben, durch HF-Modulator anschließbar an Fernseher, Blitter eingebaut; mit S/W-Monitor SM 124, Maus, Basic usw.

STE/1MByte DM 1444,-
STE/2MByte DM 1888,-
STE/4MByte DM 2333,-

HD-FLOPPY

HD-Diskettenlaufwerke im ATARI. Sie benötigen:

HD-Laufwerk 3.5", 720KB/1.44MB
TEAC FD 235 HF DM 185,-

Umschaltplatine, bestückt und getestet, 8/16MHz Umschaltung mit automatischer Erkennung intern/extern und DD/HD, keine Belastung des Shifters (!), ausführliche Einbauanleitung, HD-Formatsoftware DM 59,-

Umbaugeschick und etwas Löterfahrung oder lassen den Umbau von uns vornehmen (DM 59,-)

LASER

TEST in "TOS" 07/90

Laserdrucker für ATARI ST

ddd-LASER/8: 100% softwarekompatibel zum SLM 804. Bis 8 Seiten/Minute schnell. Geräuschlos in den Druckpausen durch Standby-Schalter. Günstig im Verbrauch. Lieferung betriebsfertig montiert. Korrekt eingestellt auf Unit 7! -Info anfordern-

ddd-LASER/8 für ATARI ST

zu günstig, Preis deshalb nur noch auf Anfrage !

Neu: Mit LCD-Digitalanzeige zur Blattzählung (Aufpreis: DM 99,-)

FESTPLATTEN

Festplatten zum ATARI ST

Die ddd-Festplatten sind vielleicht die meistgekauften "Fremd"platten, weil sie 100% kompatibel durch Verwendung des Original ATARI-Controllers (ACSI) sind. Betriebsfertig geliefert ! -Leise-

ddd-HD 32 DM 888,-
ddd-HD 48L DM 1222,-
ddd-HD 64 DM 1333,-
ddd-HD 62L DM 1377,-
ddd-HD 85 • DM 1555,-
ddd-HD 185 • DM 2888,-

• = 85 u. 185er sind nur SCSI-Platten
L = 3.5" Laufwerk = sehr leise

Bitte ausführliches Info anfordern !

COMPUTER

ST kompatibel

ddd-M1 voll ATARI kompatibel + abgesetzte Tastatur + 1 MByte Speicherausbau + aufrüstbar + mit Maus + Monitor 70Hz + zus. Anschluß für Schaltanlagen + Softwarepaket + opt. PC-SPEED + opt. 16MHz Prozessor/Cache

ddd-M1/PC DM 2098,-
ddd-M1/16MHz DM 2165,-

ddd-M2 wie vor, jedoch 2 MByte RAM wahlweise wieder mit PC-SPEED oder 16MHz

ddd-M2/PC DM 2725,-
ddd-M2/16MHz DM 2795,-

ddd-M4 wie vor, jedoch 4 MByte RAM + wahlweise mit PC-SPEED oder 16MHz

ddd-M4/PC DM zu günstig
ddd-M4/16MHz DM zu günstig

Auf Wunsch alle Rechner auch gleich mit 720/1.44MB Laufwerk ausgerüstet. Aufpreis: DM 200,-
- Bitte Info anfordern -

MULTISCAN

Der ddd Farb-Multiscan FMA 14 ist anschlußfertig für alle ATARI ST und ddd-Computer. Er erreicht eine Auflösung von 1024 • 768. Dieser ausgesuchte Monitor besticht durch eine hohe Schärfe bei kräftigen Farben und gutem Zusammenspiel mit Mega-Screen+. Mit TTL- und Analog-Eingang. Bei Systemwechsel ist der Monitor weiterhin verwendbar.

DM 1194,-

MegaScreen+

NEU: Farb-Grafikkarte für alle Mega-Rechner. Auflösung bis 832 • 624 auf Multiscan. Ideal für CALAMUS. - Info anfordern.

DM 249,-

Bei einigen Produkten kann es aufgrund knapper Zulieferungen zu Lieferzeiten kommen

Versand per NN, europaweit und Direktverkauf in Hannover

GEMISCHTES

Wechselplatte WP 44 für ATARI ST anschlußfertig DM 1777,-
Medium dazu DM 250,-

Digitalisiertablett für ATARI ST anschlußfertig DM 994,-

SCSI-Festplattensatz 85er Platte mit Controller für ATARI ST DM 1111,-

NR-KIT: (Der Dauer-Renner) Geräuschreduzierung für ATARI Festplatten, mit Software, ausführliche Anleitung, kein Löten erforderlich. DM 49,-

SOFTWARE

Adimens ST plus 3.0	DM 344,-
Arabesque	DM 219,-
Calamus	DM a.A.
Creator	DM 219,-
Laser C m. Debugger	DM 294,-
Lavadraw Plus	DM 129,-
Megamax Modula2	DM 333,-
Script	DM 188,-
Signum!2	DM a.A.
Tempus 2.05	DM 109,-
TMS Cranach	DM 349,-

... und 1000 andere Programme. Fragen Sie bitte nach. Ausführliche Programmbeschreibungen finden Sie in den Fachzeitschriften.

SCANNER

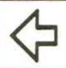
Neuer 3D-Scanner. 300 DPI Auflösung. Overhead-System. Einfach in der Handhabung. Komplett mit Software, äußerst günstig:

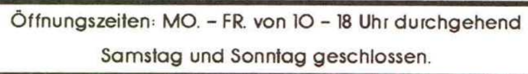
DM 1994,-

Profi-Scanner. Flachbett-Scanner mit 300/600 DPI Auflösung. 64 Graustufen. Komplett mit Scann-Software und ARABESQUE ! Reduziert im Preis:

DM 2494,-

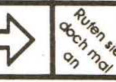
•• Geheimtip •• Wechselplattenkit: Laufwerk SyQuest SQ555 (44MB) + SCSI-Controller für ST DM 1555,-





Öffnungszeiten: MO. - FR. von 10 - 18 Uhr durchgehend

Samstag und Sonntag geschlossen.

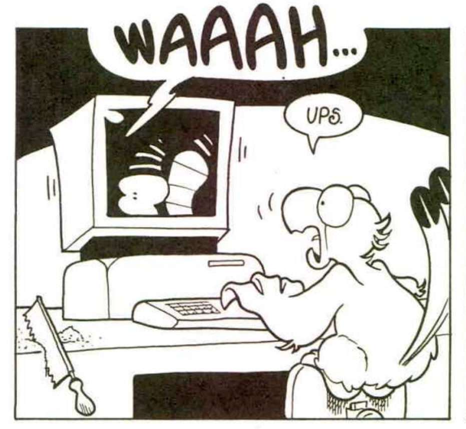


```
212:  {
213:     int fd1, fd2, i, file_pointer, null=0, wort;
214:     long flen;
215:     char byte1, byte2, byte3, byte4, symbol[9];
216:     HRD_HEADER head;
217:
218:     fd1=Fopen(from,0);  /* Datei öffnen */
219:     if (fd1<0)          /* ist nicht zu öffnen */
220:     {                   /* Abbruch */
221:        form_alert(0,FILE_ERROR1);
222:        return(-2);
223:     }
224:     fd2=Fopen(to,0);    /* Datei öffnen */
225:     if (fd2>0)          /* Datei gibt es schon */
226:     {                   /* Abbruch */
227:        Fclose(fd1);
228:        Fclose(fd2);
229:        form_alert(0,FILE_ERROR2);
230:        return(-3);
231:     }
232:     fd2=Fcreate(to,0);          /*  Schreibdatei
                                         erzeugen */
233:     head.version=1;             /*  Versionsnummer
                                         ist 1 */
234:     head.auto_name=0;           /*  keine
                          automatische Namensgebung */
235:
236:     flen= Fseek(0L,fd1,2);   /* Länge ermitteln */
237:     Fseek(0L,fd1,0);   /* an den Anfangs setzen */
238:
239:     Fread(fd1,2L,&wort);        /*  Anzahl der
                              Datensätze wird bei */
240:              /*  HRD-Format nicht verwendet */
241:     Fread(fd1,2L,&wort);        /*  Sprachflags in
                                 DEF in einem Wort */
242:     head.language=(char)wort; /*  bei HRD in
                                       einem Byte */
243:     head.auto_pos=2;  /* Zeichenausrichtung ein */
244:     head.character=1;   /* Namensgebung in GroP */
245:     head.auto_size=0;   /* keine automatische
                                    GröPenänderung */
246:     head.reserved=0;
247:
248:     Fwrite(fd2,(long)sizeof(head),&head);
249:                            /* Header schreiben */
250:     Fread(fd1, 1L, &byte1); /* Index des
                          Objektbaumes bei Objekten */
251:     Fread(fd1, 1L, &byte2); /* Index des Baumes
                              oder Objekts (Byte)*/
252:     Fread(fd1, 1L, &byte3); /* 0=Baum, 1=Objekt */
253:     Fread(fd1, 1L, &byte4);  /* Typ des Objekts */
254:
255:     Fwrite(fd2,1L,&hrd_typ[byte4]); /* Typ
                                       umkodieren */
256:     Fwrite(fd2,1L,&null);  /* reserviertes Byte
                                 auf Null setzen */
257:     wort=byte1;            /* Index des Baums
                                war Byte und wird */
258:     Fwrite(fd2,2L,&wort);        /* jetzt als
                                 Wort geschrieben */
259:     wort=byte2;          /* Index des Objekts
                                war Byte und wird */
260:     Fwrite(fd2,2L,&wort);        /* jetzt als
                                 Wort geschrieben */
261:
262:     Fread(fd1,8L,symbol);        /* Namen lesen */
263:     i=0;                       /* Zähler löschen */
264:     symbol[8]=0;  /* letzter B.stabe Stringende */
265:     do
266:     {
267:        Fwrite(fd2,1L,&symbol[i]); /* Buchstaben
                                        schreiben */
268:     }while(symbol[i++]);    /* Bis Null erreicht */
269:
270:     file_pointer=flen-16;          /* 1 Record
                             ist ja schon gelesen */
271:
272:     while(file_pointer>0)          /* Bis alles
                                 gelesen worden ist */
273:     {
274:        Fread(fd1,4L,symbol);         /* 4 Bytes
                                    überspringen */
275:        Fread(fd1, 1L, &byte1); /* Index des
                          Objektbaumes bei Objekten */
276:        Fread(fd1, 1L, &byte2); /* Index des
                          Baumes oder Objekts (Byte)*/
277:        Fread(fd1, 1L, &byte3); /* 0=Baum,
                                        1=Objekt */
278:        Fread(fd1, 1L, &byte4);
                                /* Typ des Objekts */
279:
280:        Fwrite(fd2,1L,&hrd_typ[byte4]); /* Typ
                                       umkodieren */
281:        Fwrite(fd2,1L,&null);     /* reserviertes
                            Byte auf Null setzen */
282:        wort=byte1;    /* Index war Byte und wird */
283:        Fwrite(fd2,2L,&wort);             /* jetzt
                            als Wort geschrieben */
284:        wort=byte2;      /* Index des Objekts war
                                  Byte und wird */
285:        Fwrite(fd2,2L,&wort);             /* jetzt
                            als Wort geschrieben */
286:
287:        Fread(fd1,8L,symbol);      /* Namen lesen */
288:        i=0;                     /* Zähler löschen */
289:        symbol[8]=0;    /* letzter ist Stringende */
290:        do
291:        {
292:           Fwrite(fd2,1L,&symbol[i]);
                             /* Buchstaben schreiben */
293:        }while(symbol[i++]);            /* Bis Null */
294:
295:        file_pointer-=16;    /* minus 1 Datensatz */
296:     }
297:     byte1=6;                 /* HeiPt 'Dateiende' */
298:     Fwrite(fd2,1L,&byte1);          /* schreiben */
299:     for (i=0; i<6; i++) /* Datensatz mit Nullen */
300:        Fwrite(fd2,1L,&null);         /* auffüllen */
301:     Fclose(fd1);            /* DEF-Datei schliePen */
302:     Fclose(fd2);            /* HRD-Datei schliePen */
303:     return(0);
304:  }
305:
306:  int hrd2def(from, to)   /* RCS 1.x/RSD -> Wercs
                                    konvertieren */
307:  char *from, *to;
308:  {
309:     int fd1, fd2, i, no_recs, null=0, wort1,
           wort2;
310:     long flen, langnull=0;
311:     char byte1, byte2, symbol[9];
312:     HRD_HEADER head;
313:
314:     fd1=Fopen(from,0);     /* Datei Lesen öffnen */
315:     if (fd1<0)           /* keine Datei gefunden */
316:     {                    /* -> Abbruch */
317:        form_alert(0,FILE_ERROR1);
318:        return(-2);
319:     }
320:     fd2=Fopen(to,0);  /* Datei Schreiben öffnen */
321:     if (fd2>0)           /* Datei gibt es schon */
322:     {                    /* Abbruch */
323:        Fclose(fd1);
324:        Fclose(fd2);
325:        form_alert(0,FILE_ERROR2);
326:        return(-3);
327:     }
328:     fd2=Fcreate(to,0);
329:     head.version=1;     /* Versionsnummer ist 1 */
330:     head.auto_name=0;          /*  keine
                          automatische Namensgebung */
331:
332:     flen= Fseek(0L,fd1,2);   /* Länge ermitteln */
333:     Fseek(0L,fd1,0);     /* an den Anfang setzen */
334:
335:     Fread(fd1,(long)sizeof(head),&head);
                                   /* Header lesen */
336:     wort1=0;
337:     Fwrite(fd2,2L,&wort1);       /* Anzahl der
                             Records erstmal null */
338:     wort1=(int)head.language; /* Sprachenflags,
                            vorher Byte, nun Wort */
339:     Fwrite(fd2,2L,&wort1);
340:     Fread(fd1,1L,&byte1);       /* Typ lesen */
341:     Fread(fd1,1L,&byte2);   /* unwichtiges Byte */
342:     Fread(fd1,2L,&wort1);   /* Nummer des Baums,
                                  bei Wercs Wort */
343:     Fread(fd1,2L,&wort2);     /* Objektnummer,
                                  bei Wercs Wort */
344:
345:     for(i=0; i<8; i++)         /* ganzen Namen */
```

→

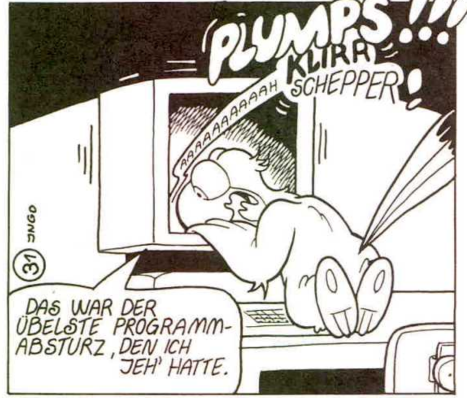
```
346:          symbol[i]=0;              /* auf Null setzen */
347:       i=0;
348:       do
349:          Fread(fd1,1L,&symbol[i]); /* Buchstabe
            lesen */
350:       while(symbol[i++]!=0);          /* bis Nullbyte */
351:
352:       if (byte1!=5)              /* Wenn Typ kein,
                                       sondern Baum */
353:          byte2=(char)wort1;      /* Nummer des Baums,
                                        bei RCS1.x Byte */
354:       else
355:          byte2=0;
356:       Fwrite(fd2,1L,&byte2);     /* schreiben */
357:       byte2=(char)wort2;         /* Objektnummer,
                                        bei RCS1.x Byte */
358:       Fwrite(fd2,1L,&byte2);     /* schreiben */
359:       byte2=(byte1==5);          /* Typ(HRD) =5
                                  entspricht Objekt =1 */
360:       Fwrite(fd2,1L,&byte2);     /* schreiben */
361:       Fwrite(fd2,1L,&def_typ[byte1]);
                     /* umkodierten Typ schreiben */
362:       Fwrite(fd2,8L,symbol);     /* Name schreiben */
363:
364:       no_recs=1;              /* Anzahl der Records */
365:       while(1)
366:       {
367:          Fread(fd1,1L,&byte1);        /* Typ lesen */
368:          if (byte1==6)                /* Wenn
                                Typ(HRD)=6=Dateieende, */
369:             break;                   /* dann raus aus
                                   der While-Schleife */
370:          Fread(fd1,1L,&byte2);        /* unwichtiges
                                          Byte */
371:          Fread(fd1,2L,&wort1);        /* Nummer des
                               Baums, bei Wercs Wort */
372:          Fread(fd1,2L,&wort2);        /* Objektnummer,
                                     bei Wercs Wort */
373:
374:          for(i=0; i<8; i++)         /* ganzen Namen */
375:             symbol[i]=0;               /* auf Null */
376:          i=0;
377:          do
378:             Fread(fd1,1L,&symbol[i]); /* Buchstabe
                                               lesen */
379:          while(symbol[i++]!=0);           /* bis
                                     Nullbyte kommt */
380:
381:          Fwrite(fd2,4L,&langnull); /* Vier Nullen
                                           schreiben */
382:          if (byte1!=5)            /* wenn es ein
                                            Baum ist */
383:             byte2=(char)wort1;      /* Nummer des
                               Baums, bei RCS1.x Byte */
384:          else
385:             byte2=0;                 /* Andernfalls
                                    auf Null setzen */
386:          Fwrite(fd2,1L,&byte2);      /* schreiben */
387:          byte2=(char)wort2;          /* Objektnummer,
                                     bei RCS1.x Byte */
388:          Fwrite(fd2,1L,&byte2);      /* schreiben */
389:          byte2=(byte1==5);           /* Typ(HRD) =5
                                  entspricht Objekt=1 */
390:          Fwrite(fd2,1L,&byte2);      /* schreiben */
391:          Fwrite(fd2,1L,&def_typ[byte1]); /*
                            umkodierten Typ schreiben */
392:          Fwrite(fd2,8L,symbol);      /* Name
                                            schreiben */
393:          no_recs++;
394:       }
395:       Fseek(0L,fd2,0);               /* wieder an den
                                     Anfangs setzen */
396:       Fwrite(fd2,2L,&no_recs);  /* Anzahl der
                                   Records schreiben */
397:
398:       Fclose(fd1);              /* DEF-Datei schliePen */
399:       Fclose(fd2);              /* HRD-Datei schliePen */
400:       return(0);
401:    }
402:
403:    int do_file(fpath,fname,srch,fpn)
404:    char *fpath,*fname,*srch,*fpn;
405:    {
406:       char tpath[64];              /* temporary path */
407:       char tname[64];              /* temporary name */
408:       int i;
409:       int key;
410:
411:       strcpy(tpath,fpath);    /* alten Pfad retten */
412:       strcpy(tname,fname);   /* alten Namen retten */
413:       if(fpath[0] == '\0')       /* wenn kein Pfad */
414:          {            /* aktuellen Pfad erstellen */
415:          fpath[0] = Dgetdrv() + 'A';   /* LW holen */
416:          strcpy(&fpath[1],":");     /* ":" anhängen */
417:          Dgetpath(fpn,Dgetdrv());     /* Pfad holen */
418:          strcat(fpath,fpn);        /* Pfad anhängen */
419:          strcat(fpath,"\\");/* Backslash anhängen */
420:          }
421:       strcat(fpath,srch);     /* Suchpfad anhängen */
422:       fsel_input(fpath,fname,&key);
                         /* Fileselektorbox aufrufen */
423:       if(key == 1)       /* OK gedrückt */
424:          {      /* Pfad und Namen zusammenkopieren */
425:          i = strlen(fpath);     /* Ende Gesamtpfad */
426:          while(fpath[i] != 0x5c)  /* Rücksuchen
                                       bis Backslash */
427:             i--;     /* also Anfang des Suchpfades */
428:          fpath[++i] = '\0';          /* abschneiden */
429:          strcpy(fpn,fpath);        /* Pfad kopieren */
430:          strcat(fpn,fname);       /* Namen anhängen */
431:          }
432:       else
433:          {
434:          strcpy(fpath,tpath);   /* Pfad zurückkop. */
435:          strcpy(fname,tname);   /* Name zurückkop. */
436:          strcpy(fpn,"\0"); /* Gesamtpfad auf Null */
437:          }
438:       return(key);             /* Taste zurückgeben */
439:    }
```

Listing 6: Konvertierung der DEF- und DFN-Datei









Alles Lesens-Werte

Experimente am Schreibtisch





Durch schnelle Computer kann seit geraumer Zeit die Simulation von Naturvorgängen realistisch vorgeführt werden.
Mit diesem Buch werden Sie in die Lage versetzt, in Ihrem ATARI ST Galaxien kollidieren zu lassen, ohne daß der Himmel einstürzt, oder gar die gewagtesten chemischen Experimente zu riskieren, ohne daß gleich das Haus in die Luft gesprengt wird.
Natürlich darf bei allem Spieltrieb die Theorie der Gesetzmäßigkeiten nicht zu kurz kommen. Ausführlich dokumentierte Listings in GFA-BASIC geben einen Einblick in die programmiertechnische Seite.

Robert Becker
Computersimulationen
337 Seiten und Diskette
DM 59,-
ISBN 3-927065-03-X

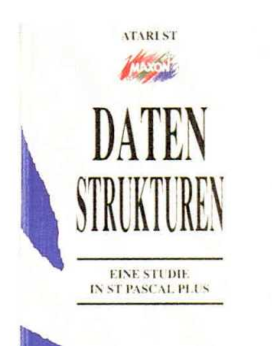
Mehr als ein Buch! Mehr als nur Software!





Mittlerweile zählt Scheibenkleister II bereits zu den Standardwerken für den ST. Die Erfolgsautoren Claus Brod und Anton Stepper beschreiben auf fast 900 Seiten alles, was man über Floppies, Festplatten, CD-ROMs und andere Massenspeicher wissen muß. Das Buch beinhaltet einen großen Kursteil, in dem Themen wie Floppy- und Festplattenprogrammierung (über Betriebssystem und direkte Controllerprogrammierung), Aufzeichnungsverfahren, Anschluß von Fremdlaufwerken und sogar Justierung und Reparaturhinweise von Laufwerken u.v.m. Zusätzlich wird eine Diskette mit 1.2 MB Software mitgeliefert, die keine Wünsche offen läßt:
Track- und Sektormonitor, bootfähige RAM-Disk, Hyperformat bringt bis zu 950 kByte, Festplattentreiber CBHD.SYS (Laden von Accessories von beliebigem Laufwerk) u.v.m.

Brod/Stepper
Scheibenkleister II
872 Seiten
und Diskette
DM 79,-
ISBN 3-927065-00-5



Fürs Programmieren unentbehrlich

Die Art und Weise wie benutzererstellte Daten behandelt werden, ob beim Speichern, beim Sortieren oder beim Suchen, ist ein zentrales Thema in der Informatik. Grund genug für Dirk Brockhaus, den verschiedenen Strukturen und Algorithmen auf mehr als 400 Seiten auf den Leib zu rücken.
Angefangen mit den Grundlagen, wie einfachste Strukturelemente, über komplexe Zahlen, sowie verschiedene Methoden für Suchen und Sortieren, bis hin zur Verarbeitung großer Datenmengen umspannt dieses Buch den gesamten Themenbereich.

Dirk Brockhaus
Datenstrukturen
403 Seiten und Diskette
DM 59,-
ISBN 3-927065-02-1

Das Buch zum Handbuch





OMIKRON.BASIC, mittlerweile als ATARI-BASIC bei jedem ST beigefügt, wird in diesem Buch ausführlich beschrieben.
Eine kurze Anleitung für den Neuling führt in die grundlegenden Kennisse der BASIC-Programmierung ein.
Es folgt die Beschreibung der Befehle, Programmierkniffe mit vielen Beispielen, sowie Aufgaben und Lösungen.
Sprites werden erklärt, die Overlaytechnik zeigt das Auslagern langer Programmteile nebst Laden bei Gebrauch.
Was sind Bibliotheken (Libraries)? Das Mitbenutzen fremder Bibliotheksfunktionen, Aufrufe von TOS und GEM, sowie GDOS wird kein Geheimnis bleiben.
Ein weiterer Schwerpunkt ist die Sound- und Grafikprogrammierung unter OMIKRON-Basic, mit Klarheit über Metafiles, IMG-Format usw. Als Spezialität folgt die Beschreibung von internen Multitasking-Befehlen.

Clemens Hoffmann
OMIKRON.BASIC 3.0
355 Seiten
und Diskette
DM 59,-
ISBN 3-927065-01-3



Bestellcoupon MAXON Computer GmbH Schwalbacherstr. 52 6236 Eschborn Tel.: 06196/481811

Name/Vorname

Straße

Ort

Unterschrift

Hiermit bestelle ich:

☐ SCHEIBENKLEISTER II DM 79,00

☐ COMPUTER-SIMULATIONEN DM 59,00

☐ OMIKRON.BASIC 3.0 DM 59,00

☐ DATENSTRUKTUREN DM 59,00

Versandkosten: Inland DM 7,50
Ausland DM 10,00

Auslandbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr

☐ Vorauskasse
☐ Nachnahme

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3
Vertrieb in Frankreich: AROBACE 2 Rue Piemontesis F-75018 Paris Tel: 1/42235044 Fax 1/42545631

Somewhere over the rainbow

Untere Ebene der Dateiverwaltung

4

Untere Ebene der Dateiverwaltung Heute geht es um die untere Ebene der Dateiverwaltung; vor allem um die Sektorpufferung und die FAT-Verwaltung, bei denen es erhebliche Verbesserungen gegeben hat, was vor allem die Harddisk-Benutzer zu schätzen gelernt haben.

Sektorpufferung - was war das noch?

Zunächst möchte ich in wenigen Sätzen das Wesentliche der GEMDOS-Sektorpufferung (von ATARI "GEMDOS Cache" genannt) zusammenfassen, da dies doch schon länger her ist (März 1988, [1]). Bevor die Experten zur nächsten Überschrift springen, sei darauf hingewiesen, daß eine Neuigkeit in diesem Abschnitt versteckt ist - also weiterlesen oder den Pfeil '—>' suchen.

Für jedes "logische Laufwerk" (unter GEMDOS mit A:, B:, C:, ... bezeichnet) ist ein Laufwerkstreiber zuständig, der über BIOS- Funktionen ('Rwabs', 'Getbpb', 'Medch') vom GEMDOS aufgerufen wird. Der Floppy-Laufwerkstreiber ist im BIOS integriert, alle anderen müssen extern geladen werden (Harddisk-, RAM-Disk-, EPROM-Disk-Treiber).

Die Massenspeichermedien sind in "Sektoren" unterteilt, auf die GEMDOS mittels 'Rwabs' zugreifen kann. Wie diese Sektoren physikalisch auf dem Medium realisiert sind, geht nur den Laufwerkstreiber etwas an, daher werden sie oft auch als "logische Sektoren" bezeichnet, um sie von den "physikalischen Sektoren", in die Disketten und Harddisks unterteilt sind, zu unterscheiden. Da GEMDOS nur "logische Sektoren" kennt, werde ich hier nur von "Sektoren" sprechen. Die Größe der logischen Sektoren eines Mediums muß einheitlich sein, mindestens 256 Bytes betragen und eine Zweierpotenz sein (üblich ist 512).

GEMDOS faßt nun einige Sektoren zu einem "Cluster" zusammen, der immer nur als Ganzes einer Datei zugeordnet wird. Die Zahl der Sektoren pro Cluster muß dabei eine Zweierpotenz sein (üblich ist 2).

Den genauen Aufbau eines Mediums erfährt GEMDOS mittels der BIOS-Funktion 'Getbpb', die einen Zeiger auf die Struktur "BIOS Parameter Block" (BPB) liefert (Abb. 1). Auf jedem Medium werden vom Laufwerkstreiber einige Sektoren für ein Wurzelverzeichnis ("Root Directory") und eine FAT (File Allocation Table) reserviert. Ein Eintrag der FAT gehört dabei zu einem Cluster und gibt an, zu welcher Datei der Cluster gehört bzw. ob er frei ist. Ein Eintrag ist entweder 16 oder 12 Bits breit (ebenfalls im BPB festgelegt).

Um Massenspeicherzugriffe zu sparen, werden gewisse, einmal gelesene Sektoren vom GEMDOS gepuffert. Dazu existiert zu jedem gepufferten Sektor ein "Buffer Control Block" (BCB), der Angaben über den Sektor und die Daten im Puffer macht (Abb. 2). Diese BCBs sind in zwei Pufferlisten angeordnet.

—> [Experten-Weiterlese-Stelle]

Und hier sind wir auch schon bei der ersten Neuerung von TOS 1.4. In der Liste 0 sind nämlich alle BCBs für FAT- und Wurzelverzeichnis-, in der Liste 1 alle für Daten-Sektoren (einschließlich Unterverzeichnisse) miteinander verkettet. Bisher enthielt Liste 1 auch die Verzeichnis- und demnach Liste 0 nur die FAT-Sektoren! Die jetztige Anordnung ist von ATARI in der TOS1.4-Dokumentation korrekt beschrieben und entspricht damit sogar einer Uralt-ATARI-Doku (wenn ich mich recht erinnere dem "Hitchhiker's Guide to the BIOS"). In der Praxis macht dies nicht viel aus, da die Wurzelverzeichnis-Sektoren sowieso in einer hoffnungslosen Minderheit gegenüber den anderen sind. Die Listen sind so organisiert, daß die Sektoren in der Reihenfolge des letzten Zugriffs sortiert sind. Der erste Sektor ist der zuletzt angesprochene usw.

Verwaltung der Pufferlisten

Zunächst einmal die gute Nachricht für all die, die an den nachfolgenden Details weniger interessiert sind. Die Sektorpufferung im TOS 1.4 funktioniert endlich! Die in [1] beschriebenen Haken und Ösen sind beseitigt. Von ATARI gibt es sogar ein Programm, das die Pufferlisten erweitert.

Die Verwaltung der Puffer sowie das Lesen und Schreiben von Sektoren geschieht mit einigen internen GEMDOS-Routinen, die ich wie in [1] benennen möchte. Es wird noch einmal kurz die Arbeitsweise zusammengefaßt, um dann die Änderungen beim "neuen GEMDOS" herauszustellen. Das Lesen eines einzelnen Sektors über die Pufferliste (auch bei Schreibzugriffen auf einen Teil des Sektors notwendig) besorgt 'f_sread'.

Wenn der Sektor noch nicht gepuffert ist, wird er in einen freien Puffer der jeweiligen Liste geladen. Ist keiner mehr unbenutzt, wird der "älteste" Puffer (also der letzte) aus der Liste entfernt. Dieser wird dabei natürlich zurückgeschrieben, falls er geändert wurde. Der angeforderte Sektor wird nun in den freien bzw. freigemachten Puffer geladen (mit 'Rwabs'). Falls hierbei ein Fehler auftritt, wird die gesamte GEMDOS-Funktion abgebrochen, wie in [2] besprochen. Während

dieser Fall an vielen anderen Stellen des GEMDOS nur unzulänglich behandelt wird, gab es hier eigentlich nichts zu meckern. Doch im neuen GEMDOS wird vor dem 'Rwabs'-Aufruf der 'bufrec'-Wert des BCB auf -1 gesetzt, wohl um ihn "ungültig" zu machen.

Einmal abgesehen davon, daß dies normalerweise durch einen 'bufdrv'-Wert von -1 gemacht wird (hier dürfte wohl ein Tippfehler zugeschlagen haben), ist dies auch völlig überflüssig, ja sogar schädlich. Denn wenn ein Lesefehler auftritt, bleibt der so verunstaltete BCB belegt, obwohl mit einer ungültigen Sektornummer versehen. Dies jedoch nicht für immer, denn irgendwann wird er als "nicht mehr so wichtig, da hinten in der Liste" erkannt und erneut freigemacht. Glücklicherweise kann er dann nicht verändert worden sein, so daß es keinen Schreibversuch auf eine "verrückte" Sektornummer (d.h. -1) gibt. Dieser neue GEMDOS-Fehler bleibt also ohne Folgen. Ansonsten arbeitet das bis dato fehlerlose 'f_sread' wie bisher, d.h., bei einem geplanten Schreibzugriff auf den Sektor wird das 'dirty'-Flag im BCB gesetzt.

Die Routine 'f_swrite' dient dem Zurückschreiben eines eventuell veränderten Sektors. Sie wird z.B. aus 'f_sread' (wie gerade erwähnt) oder beim Schließen einer Datei aufgerufen, um das Speichermedium wieder auf den gleichen Stand wie den Puffer zu bringen.

Der bisherige Fehler, daß unveränderte Puffer ('dirty' gleich Null) ungültig gemacht wurden, ist beseitigt. Nun passiert bei unveränderten oder ungültigen Puffern einfach nichts, wie es sein sollte. Dieser Fehler war einer der Gründe für die faktische Wirkungslosigkeit der Sektorpufferung.

Auch hier gibt es eine merkwürdige Änderung, die mit einem neuen Fehler verbunden ist. Und zwar wird nun die entsprechende Pufferliste nach dem zu schreibenden Sektor durchsucht. Es werden alle veränderten Puffer der Liste geschrieben, die dem gleichen Laufwerk und gleichen Sektor des eigentlich zu schreibenden Sektors zugeordnet sind. Da jeder Sektor jedoch nur einen Puffer haben kann, scheint mir dies darauf hinauszulaufen, daß in der Liste eh immer genau der Sektor gefunden wird, dessen BCB schon vorher bekannt war. Somit halte ich diese Änderung für ziemlichen Quatsch.

Der Fehler besteht nun darin, daß der Anfang von 'bufl[]', das den Zeiger auf die jeweilige Liste beinhaltet, als erster BCB aufgefaßt wird (hier steht im Quelltext ein Adreß-Operator zuviel). Da dort im allgemeinen nur "Müll" steht, der nicht die drei Bedingungen ('b_bufdrv', 'b_buftyp', 'b_dirty' müssen stimmen) erfüllt, die überprüft werden, bevor ein Sektor geschrieben wird, hat dies glücklicherweise keine nachteiligen Konsequenzen. Hier wurde anscheinend "ähnlicher" Quelltext von anderer Stelle abgeschrieben, an der der Adreß-Operator korrekt ist.

```
typedef struct
{ int b_recsiz;  /* Bytes pro Sektor                              (512) */
  int b_clsiz;   /* Sektoren pro Cluster                            (2) */
  int b_clsizb;  /* Bytes pro Cluster                            (1024) */
  int b_rdlen;   /* Sektoren für Root Directory                     (7) */
  int b_fsiz;    /* Sektoren pro FAT                                (5) */
  int b_fatrec;  /* erste Sektornummer der zweiten FAT              (6) */
  int b_datrec;  /* Sektornummer des ersten Daten-Clusters         (18) */
  int b_numcl;   /* Anzahl Daten-Cluster auf Diskette             (711) */
  int b_flag[8]; /* Flags (Bit 0 von b_flag[0]: FAT-Typ)            (0) */
} BPB;

(Werte für Standarddiskette in Klammern)
```

Abb. 1: Struktur des BIOS Parameter Blocks (BPB)

```
typedef BCB
{ BCB *b_link;    /* Zeiger auf nächsten BCB dieser Liste           */
  int b_bufdrv;   /* Laufwerksnummer, -1 für ungültigen BCB         */
  int b_buftyp;   /* FAT (0), DIR (1), DATA (2)                     */
  int b_bufrec;   /* Sektor-Nummer in GEMDOS-Zählung                */
  int b_dirty;    /* ungleich Null: Puffer-Inhalt geändert          */
  DMD *b_dmd;     /* Zeiger auf DMD von b_bufdrv                    */
  char *b_bufr;   /* Zeiger auf eigentlichen Sektor-Puffer          */
} BCB;
```

Abb. 2: Struktur des Buffer Control Blocks (BCB)

Die interne Funktion 'f_mrw' übernimmt die Übertragung mehrerer zusammenhängender Sektoren direkt über 'Rwabs' unter Umgehung der Pufferliste. Hier müssen Kollisionen mit der Pufferliste 1 vermieden werden. Der gravierende Fehler, der zu Datenverlusten führen konnte (Beispiel-Listing in [1]), ist behoben, allerdings nicht optimal. Es wird nun prinzipiell jeder Sektor in der Pufferliste 1 zurückgeschrieben und ungültig gemacht, wenn er von 'f_mrw' übertragen werden soll. Ausreichend und optimaler wäre es gewesen, wenn er bei einem Schreibzugriff ungültig gemacht und bei einem Lesezugriff zurückgeschrieben würde.

Aber naja, immerhin funktioniert die Sektorpufferung nun überhaupt; und das, obwohl zwei alte gegen zwei neue Fehler eingetauscht wurden, die allerdings harmloser Natur sind.

Größere Cluster!

Kommen wir nun noch kurz zu den übergeordneten Routinen der Dateiverwaltung. In [2] wurde die Routine 'f_frw' besprochen, die ähnlich wie die GEMDOS-Funktionen 'Fread' und 'Fwrite' beliebige Teile einer Datei lesen und schreiben kann. Sie erledigt auch die Umsetzung von logischen Dateipositionen in Cluster- und dann in Sektornummern, die von den eben behandelten Funktionen verstanden werden, und ist im Wesentlichen unverändert.

Hier wurde aber der Fehler korrigiert, der mehr als zwei Sektoren pro Cluster verhinderte. Der von mir in [2] erwähnte und nicht funktionierende Patch hatte einen Fehler, der mir erst aufging, als ich die "offizielle" Version im neuen 'f_frw' entdeckte. Da 4 Sektoren pro Cluster bei einer provisorisch gestrickten RAM-Disk problemlos funktionieren, gehe ich davon aus, daß es keine weiteren Fallstricke hierfür gibt.

In der Praxis sind mehr Sektoren pro Cluster deswegen problematisch, da dies bei alten TOS-Versionen nicht möglich, und außerdem die Anzahl der Sektoren unabhängig von der Cluster- Zahl begrenzt ist (s. Diskussion unten). Daher hat ATARI beim Harddisk-Treiber AHDI den Weg zu größeren logischen Sektoren eingeschlagen und umgeht somit die Beschränkungen des GEMDOS.

FAT-Verwaltung

Die Handhabung der FAT habe ich früher immer recht kurz abgehandelt. Dies lag daran, daß die FAT intern fast genau wie eine Datei behandelt wurde. Dies ging soweit, daß auch Zugriffe auf die FAT mit den internen Versionen von 'Fread' und 'Fwrite' erledigt wurden, was zwar auf den ersten Blick recht einleuchtend scheint und code-sparend ist, auf den zweiten Blick aber der Hauptgrund für die Ineffizienz der FAT-Verwaltung war. Einen Eindruck bekommt man bei Programmen wie WORDPLUS, die Dateien byte-weise mit 'Fread' lesen; nahezu genauso lahm war auch das Lesen der FAT.

Wie jedem TOS 1.4- und Harddisk-Benutzer wohl mit als erstes aufgefallen sein dürfte, zeichnet sich das neue GEM-

DOS durch eine enorme Beschleunigung der Harddisk-Operationen aus, vor allem beim Ermitteln des freien Speicherplatzes ('Dfree') und Anlegen von Dateien ('Fcreate'). Dies ist fast ausschließlich einer im Kern zwar noch gleichen, aber in einigen neuralgischen Punkten verbesserten FAT-Verwaltung zu verdanken.

Es gibt eine neu Routine, die eigens für das Lesen von FAT- Sektoren zuständig ist, von mir 'f_qread' genannt. Sie liest einen FAT-Sektor mit 'f_sread' über die Pufferliste und merkt sich alle Angaben, die zu seiner eindeutigen Charakterisierung notwendig sind, in neuen globalen Variablen ["Drive Media Descriptor" (DMD), Sektornummer, Pufferadresse und "Schreib-Flag", s. Abb. 3]. Der DMD gibt dabei das Laufwerk an, und das Schreib-Flag zeigt an, ob der Puffer demnächst verändert werden soll (wie bei 'f_sread'). Außerdem gibt ein Flag 'qf_invalid' an, ob diese Daten gültig sind (1 für ungültig, 0 für gültig). Zunächst einmal machen die beiden Routinen, die für das Lesen ('f_fatrd') bzw. Schreiben ('f_fatwr') eines FAT-Eintrags zuständig sind, hiervon Gebrauch, indem sie vor einem Aufruf von 'f_qread' prüfen, ob 'qf_invalid' gelöscht ist sowie 'qf_dmd' und 'qf_rec' mit den gewünschten Werten übereinstimmen. Wenn ja, kann mittels 'qf_buf' direkt auf den gepufferten FAT-Sektor zugegriffen werden. Bei 'f_fatwr', wo ein FAT-Sektor beschrieben werden soll, muß 'f_qread' allerdings auch aufgerufen werden, wenn 'qf_write' gelöscht ist, damit das 'b_dirty'-Flag im BCB von 'f_sread' gesetzt werden kann.

Die dritte elementare FAT-Routine ('f_nxtcl'), die den nächsten Cluster einer Datei bzw. einen freien Cluster ermittelt sowie eine Datei automatisch verlängert, wenn über ihr Ende hinausgeschrieben werden soll, macht selbst keine eigenen FAT- Zugriffe, sondern benutzt hierfür 'f_fatrd' und 'f_fatwr'.

Obwohl also gar keine Massenspeicherzugriffe eingespart werden, ergibt sich eine erhebliche Effizienzsteigerung, da praktisch der gesamte Überbau an internen Routinen übergangen wird (früher wurde einfach ein 'Fread' bzw. 'Fwrite' auf eine "FAT-Datei" gemacht).

Nun scheint dieses Verfahren aber zunächst ziemlich waghalsig zu sein, da die in den 'qf_'-Variablen festgehaltenen Angaben rasch ungültig werden können (Sektorpuffer wird freigegeben oder zurückgeschrieben, Mediumwechsel). Daher wird in allen Routinen, die eine der drei FAT-Routinen benutzen (meist in einer Schleife, wo sonst "nicht viel passiert") vor dem ersten Aufruf 'qf_invalid' gesetzt, so daß beim ersten Mal für "klare Verhältnisse" gesorgt wird. Übrigens wird auch nach den FAT-Operationen jeweils 'qf_invalid' gesetzt, was zwar nicht notwendig ist, aber wohl der Sicherheit dienen soll. In diese Kategorie "doppelt hält besser" fällt auch das Verhalten von 'f_qread', welches vor dem 'f_sread'-Aufruf ebenfalls 'qf_invalid' setzt, damit bei einem Lesefehler, der zum Abbruch der GEMDOS-Funktion führt, die 'qf_'-Variablen ungültig sind. Ich hatte zwar zuerst erhebliche Bedenken, ob dies wirklich "wasserdicht" ist, und bin mir immer noch nicht ganz sicher, aber es scheint gut zu funktionieren.

```
DMD *qf_dmd;      /* $3CDE: Zeiger auf DMD zu 'qf_buf'                        */
int qf_rec;       /* $3BA6: Sektornummer zu 'qf_buf'                          */
char *qf_buf;     /* $5E92: Zeiger auf FAT-Sektorpuffer                       */
char qf_invalid;  /* $3CE2: =TRUE wenn andere 'qf_'-Variablen ungültig */
char qf_write;    /* $3BA8: =TRUE wenn 'qf_buf' geändert                      */
```

Abb. 3: GEMDOS-Variablen zur "schnellen FAT"

Die Routinen bzw. GEMDOS-Funktionen, die diese FAT-Routinen aufrufen, und bei denen daher mit einer deutlichen Geschwindigkeitssteigerung zu rechnen ist, sind im einzelnen: 'f_frw' beim Übertragen von "Sektorgruppen", 'Dfree', 'Fsnext', internes 'f_fcreate' (bei 'Fcreate', 'Dcreate'), internes 'f_fdelete' (bei 'Fdelete' für Dateien), internes 'f_fseek' (bei 'Fseek' sowie anderen GEMDOS-Funktionen).

Auch bei dieser Änderung des GEMDOS hat sich eine Merkwürdigkeit eingeschlichen, die ich allerdings noch weniger verstehe. Wie oben erklärt, werden in 'f_fatrd' und 'f_fatwr' die 'qf_'- Variablen abgefragt, um zu entscheiden, ob 'f_qread' aufgerufen werden muß. Bei Schreibzugriffen ist dies wie gesagt auch dann notwendig, wenn 'qf_write' gelöscht ist, bei Lesezugriffen ist der Wert 'qf_write' unerheblich. Trotzdem ist in 'f_fatrd' an allen drei Stellen eine Abfrage vorhanden, die zum Aufruf von 'f_qread' führen würde, wenn 'qf_write' gesetzt ist (also genau umgekehrt wie im "Schreibfall", aber unnötig). Doch ist im Assembler-Code vorher ein Sprungbefehl, der um diese Abfrage herumführt, so daß sie gar nicht ausgeführt werden kann! So etwas kann man in C nur mit einem 'goto' formulieren, welches aber andererseits in einem '?:'-Ausdruck, der hier vorliegt, nicht stehen kann. Mir ist völlig rätselhaft, wie man so etwas in C überhaupt formulieren kann, geschweige denn warum. Ein Compiler- Fehler scheint ebenfalls recht unwahrscheinlich zu sein.

In der BETA-Version des TOS 1.4 vom Frühjahr 1988 war dies die einzige wesentliche Verbesserung der FAT-Verwaltung, und das war schon äußerst wirkungsvoll. Doch die "endgültige" Version weist noch mehr positive Überraschungen auf. So gibt es eine weitere neue Routine, die speziell für das Suchen des ersten freien Clusters bzw. das Zählen aller freien Cluster ausgelegt ist. Sie wird von 'f_nxtcl' aufgerufen, wenn der erste Cluster einer Datei ermittelt werden muß (also nicht beim echten Verlängern von Dateien), und von 'Dfree' benutzt.

Im Gegensatz zur "schnellen FAT" von oben kann diese Routine nur 16-Bit-FATs (i.a. also bei Hard- und RAM-Disks) bearbeiten. Sie liest ganz einfach der Reihe nach FAT-Sektoren mit 'f_qread' und durchsucht den Sektorpuffer in gut programmierten Schleifen direkt nach freien Clustern. Bei 12-Bit-FATs wird die FAT ganz "konventionell" mit 'f_fatrd' gelesen.

Über FAT-Einträge

Nach diesem Ausflug in eine ATARI-untypische Programmierwut lassen Sie mich wieder zu den "Kleinigkeiten" zurückkehren, die es ja auch in sich haben können, hier speziell zur den Tücken der FAT-Verwaltung, was "fehlerhafte" Cluster angeht.

Gültige Cluster-Nummern beginnen bei 2 und gehen bis $7FF (12- Bit-FAT) bzw. $7FFF (16-Bit-FAT). $FFF bzw. $FFFF steht für den letzten Cluster einer Datei (kurz EOF genannt), und $FF0-$FFE bzw. $FFF0-$FFFE sind (zumindest bei GEMDOS' großem Bruder MS-DOS) für Sonderzwecke vorgesehen (hier kurz als "Fehler-Cluster" bezeichnet). Der Rest ab $800 bzw. $8000 wird bzw. wurde nicht benutzt (ich nenne so etwas verschwendet), siehe auch den nächsten Abschnitt. Die gültigen Cluster-Nummern wurden schon ab TOS 1.2 richtig ausgewertet, allerdings gab es noch einigen Kuddelmuddel bei den "Fehler-Clustern". Nun ist es auf jeden Fall so, daß die elementare FAT-Routine 'f_fatrd' die Cluster-Nummern einheitlich und korrekt behandelt.

Bei 16-Bit-FATs wird der FAT-Eintrag unverändert an den Aufrufer zurückgegeben, bei 12-Bit-FATs werden alle Einträge der Form xFFx auf die Form FFFx gebracht und alle anderen zu 0xxx gemacht. Damit haben EOF und Fehler-Cluster für übergeordnete Routinen das gleiche Format (nämlich $FFFF bzw. $FFFx), unabhängig vom FAT-Format. Gültige Cluster-Nummern sind immer

positiv ($0000 bis $07FF bzw. $7FFF).

Soweit, so gut. Die Auswertung dieser Information an "höherer Stelle" läßt jedoch, wie auch schon früher, zu wünschen übrig. Beim Ermitteln des nächsten Clusters mit 'f_nxtcl' oder beim internen 'Fseek' wird nur bei EOF abgebrochen. Fehler-Cluster werden als Cluster-Nummern akzeptiert! Daher sind "negative" FAT-Einträge (außer $FFF bzw. $FFFF) weiterhin verboten. Beim internen 'Fdelete' dagegen wird das Löschen der Datei auch bei Fehler-Clustern abgebrochen. Dies war auch schon früher so, funktionierte damals aber nur bei 16-Bit-FATs.

Bisher wurde an dieser Cluster-Nummern-Geschichte noch bei jeder TOS-Version etwas geändert, auch zwischen den beiden Beta- Versionen und dem jetzigen TOS 1.4 gibt es Unterschiede. Ich bin mal gespannt, wieviele Jahre noch vergehen werden, bis ATARI dies geregelt bekommt. So wäre es auch problemlos möglich, die Cluster-Nummern $800-$FEF bzw. $8000-$FFEF zuzulassen, vergleiche dazu die Ausführungen im nächsten Abschnitt.

Sektorzählung und Mediumgröße

In [3] bin ich ausführlich auf die verschiedenen Zählweisen für Sektoren und Cluster von BIOS und GEMDOS eingegangen. Eine Übersicht zeigt Abb. 4.

Für FAT- bzw. Wurzelverzeichnis werden Pseudo-Cluster-Nummern generiert, die von 'fst' bis 'rdst'-1 bzw. 'rdst' bis -2 laufen. 'fst' und 'rdst' errechnen sich dabei aus den Laufwerksparametern im BPB. Daten-Cluster haben Nummern von 2 bis maximal 32767, die Cluster-Nummern -1, 0 und 1 sowie die übrigen negativen Nummern von -32768 bis 'fst'-1 sind unbenutzt. Die GEMDOS-Sektornummern ergeben sich durch Multiplikation mit der Anzahl der Sektoren pro Cluster und werden ihrerseits mit drei konstanten Offsets für die drei Sektortypen in die von 'Rwabs' verstandenen, stets positiven Sektornummern umgerechnet. Die GEMDOS-Sektornummern für Datensektoren müssen dabei ebenfalls positiv sein, so daß der Bereich der Cluster-Nummern evt. noch weiter eingeschränkt wird (bei den üblichen 2 Sektoren pro Cluster also auf 2 bis 16383). An diesem Prinzip hat sich nichts geändert, insbesondere die Formeln in Abb. 4 gelten weiterhin. Neu ist, daß nur die negativen Sektor- und Cluster-Nummern reserviert sind, die auch tatsächlich benötigt werden. Dies sind die Cluster-Nummern 'fst' bis -1, die anderen negativen Nummern werden nun als positive (d.h. vorzeichenlose) Zahlen angesehen. Es sind also nun größere Nummern erlaubt, wenn auch die Obergrenze von der Größe von FAT und Wurzelverzeichnis abhängt.

Zumindest ist dies offenbar die Absicht gewesen, die hinter einigen Änderungen steckt. Die Realisierung läßt allerdings zu wünschen übrig. Es gibt drei uns wohlbekannte Routinen, die daraufhin geändert wurden.

Bei 'f_sread' wird an Hand der GEMDOS-Sektornummer entschieden, ob es sich um einen FAT-, Wurzelverzeichnis- oder Datensektor handelt. Dort gibt es nun eine recht kompliziert geratene Abfrage, die aber falsch programmiert ist, so daß zwar die Abfrage als solche stimmt, aber Datensektornummern nicht größer als 'fst' - 1 - 2 * fatrec sein dürfen (statt 'fst' - 1), wobei 'fatrec' die erste BIOS- Sektornummer der "zweiten" FAT ist ('b_fatrec'-Wert aus dem BPB, bei Standarddisketten also 6). Dies ist aber nicht weiter schlimm, da man das "obere Ende" der Sektornummern sowieso nicht ausreizen sollte (da es vom Medium abhängt).

In der Routine 'f_fatrd' gibt es eine Abfrage, die (korrekt!) Wurzelverzeichnis-Cluster-Nummern herausfischt, aber bei FAT- Clustern versagt. Da aber die FAT nicht mehr wie eine Datei behandelt, sondern separat verwaltet wird, macht dies auch nichts. Bei 'f_nxtcl' schließlich ist alles danebengegangen.

Verzeichnis-Cluster werden zunächst für Daten-Cluster gehalten, was aber Dank des später aufgerufenen 'f_fatrd' wieder ins Lot kommt. Dagegen werden "negative" Daten-Cluster stets als FAT-Cluster angesehen, was zu einer echten Fehlfunktion führt: Als "Folge-Cluster" eines Clusters wird nicht der FAT-Eintrag, sondern einfach der mit der nächsthöheren Cluster-Nummer genommen. Da aber die FAT-Verwaltung zur Zeit sowieso nur positive FAT- Einträge zuläßt, ist auch dies letzlich egal.

Das Fazit lautet: Größere Sektornummern sind nun tatsächlich möglich, so daß man auf fast 32 MB statt bisher 16 MB große Harddisk-Partitionen kommen kann (bei 1024 Byte großen Clustern), größere Cluster-Nummern jedoch nicht. Allerdings ist auch ATARI hier zurückhaltend und garantiert in seiner neuesten Dokumentation zum Harddisk-Treiber nur 32768 Sektoren. Vermutlich sind sich die Programmierer darüber im Klaren, daß die Numeriererei noch nicht das Gelbe vom Ei ist.

Was ich hierbei überhaupt nicht verstehe, ist, warum GEMDOS intern nicht einfach auf 'longs' (32-Bit-Integer) umsteigt, dann gibt es genug positive Nummern. Die FAT-Einträge könnten ja weiterhin 16 Bits breit bleiben (aus Kompatibilitätsgründen), da hier nur Daten-Cluster-Nummern stehen. Einziger Nachteil wäre die Änderung der von ATARI dokumentierten BCB-Struktur, da bestehende Pufferlistenerweiterungsprogramme nicht mehr laufen würden. Zur Einführung des TOS 1.4 wäre dies aber noch problemlos möglich gewesen!

0	bis	1	unbenutzt
2	bis	32767	Daten-Cluster
-32768	bis	'fst'-1	Daten-Cluster (mit Vorbehalten), früher unbenutzt
'fst'	bis	'rdst'-1	FAT-Cluster
'rdst'	bis	-2	Wurzelverzeichnis-Cluster
-1			unbenutzt

Dabei sind (Division jeweils ohne "Rest"):

rdcl = (rdlen + clsiz - 1) / clsiz
rdst = -1 - rdcl
fcl = (fsiz + clsiz - 1) / clsiz
fst = -rdst - fcl

mit folgenden Werten aus dem BPB:

clsiz	Sektoren pro Cluster
rdlen	Sektoren für Wurzelverzeichnis
fsiz	Sektoren pro FAT

Abb. 4: Interne Cluster-Numerierung unter GEMDOS

Das Wort zum Dienstag

(Heute ist Montag, kurz vor Redaktionsschluß). So, das war diesmal zwar kürzer als sonst, aber dafür habe ich das neue GEMDOS komplett neu rekompiliert, was mir ATARI nicht gerade leicht gemacht hat, da insbesondere in der Dateiverwaltung alle Routinen bunt durcheinandergewürfelt sind (gegenüber dem alten GEMDOS). Die innere Logik der Anordnung ist zwar nicht besser geworden, aber die ATARI-Programmierer mochten offenbar eine andere Strukturierung als die Digital Research- Programmierer. Nun, dafür kann ich dann die nächsten Male aus dem Vollen schöpfen.

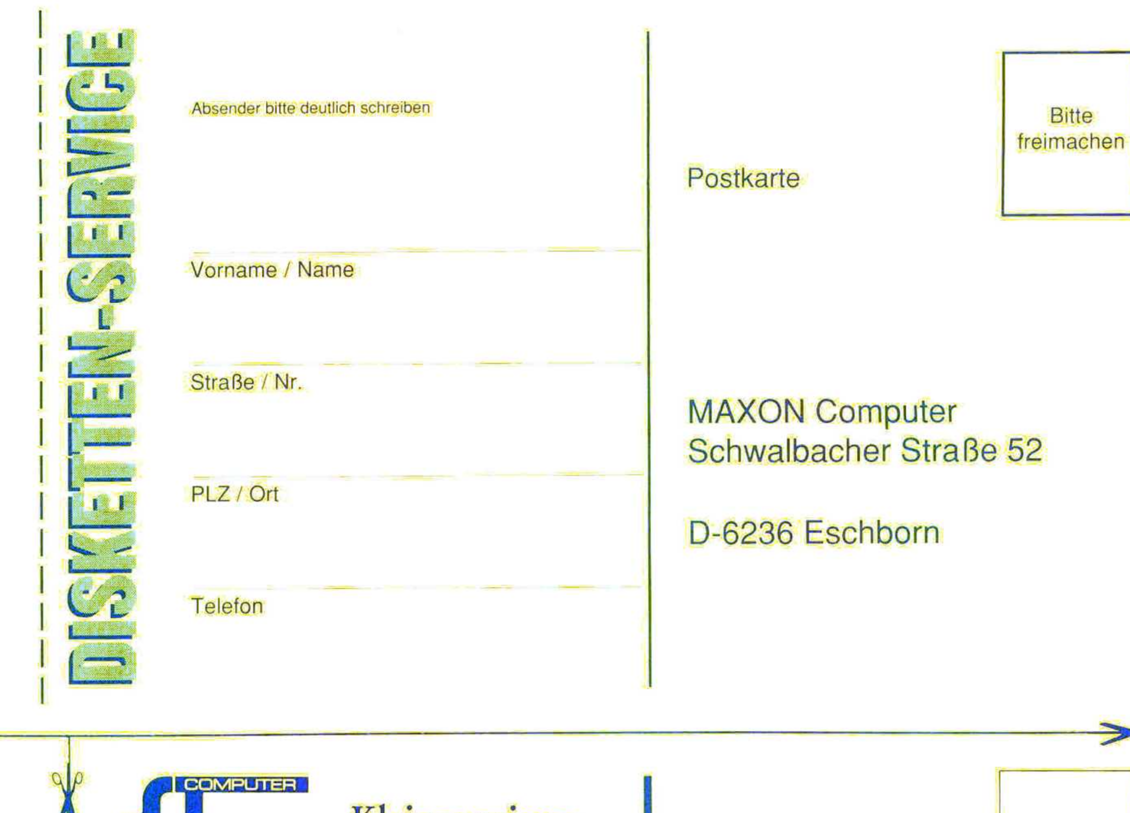
Alex Esser

Literatur:

[1] A.Esser: "Auf der Schwelle zum Licht - Dateizugriff auf Massenspeicher", ST-Computer 3/88

[2] A.Esser: "Auf der Schwelle zum Licht - Teil 1", ST-Computer 12/87

[3] A.Esser: "Auf der Schwelle zum Licht - Laufwerksverwaltung", ST-Computer 10/88



DISKETTEN-SERVICE

Absender bitte deutlich schreiben

Vorname / Name

Straße / Nr.

PLZ / Ort

Telefon

Postkarte

Bitte freimachen

MAXON Computer
Schwalbacher Straße 52

D-6236 Eschborn

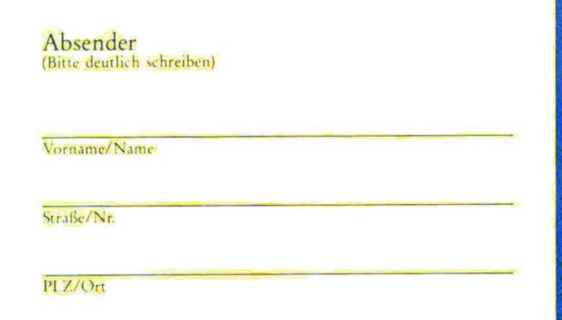


Kleinanzeigen



Kleinanzeigen





Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Heim Verlag

Heidelberger Landstr. 194

6100 Darmstadt 13







ABO



Postkarte

Absender = Rechnungsadresse
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Heim Verlag

Heidelberger Landstr. 194

6100 Darmstadt 13



Bitte senden Sie mir folgende PD-Disketten / Sonderdisks

Datum / Unterschrift

Zahlung erfolgt:

___ Vorauskasse (per Scheck) ___ per Nachnahme

- Pro PD-Diskette entfällt ein Unkostenbeitrag von DM 10,-
- Preis der Sonderdisks je nach Angabe
- Bei Nachnahme zuzüglich DM 4,- Nachnahmegebühr
- Versandkosten (Porto und Verpackung) DM 5,- (Ausland DM 10,-)
- Auslandsbestellungen nur gegen Vorauskasse möglich

Ab 5 Disketten versandkostenfrei

SONDERDISK

Bitte beachten Sie, daß Sonderdisks ein Copyright besitzen und nicht weiterkopiert werden dürfen.

DISKETTEN-SERVICE



Kleinanzeigen-Auftrag

Bitte veröffentlichen Sie für mich folgende Kleinanzeige in der angekreuzten Rubrik

Biete an ☐ Hardware ☐ Software **Ich suche** ☐ Hardware ☐ Software ☐ **Tausch** ☐ **Kontakte** ☐ Verschiedenes

30 Buchstaben je Standardzeile – incl. Satzzeichen und Wortzwischenräume.
Groß- und Kleinbuchstaben verwenden, fettgedruckte Wörter unterstreichen.

Bearbeitung **nur gegen Vorausscheck** über den entsprechenden Betrag (keine Überweisung)

☐ privat = DM 4.- je Zeile
☐ gewerblich = DM 12.- je Zeile
☐ Chiffregebühr = DM 10.-

Scheck über DM ______ ist beigefügt

Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

Datum Unterschrift

Absenderangaben auf der Rückseite nicht vergessen

COMPUTER

Kleinanzeigen





Abonnement

Kd.-Nr.:

Ja, bitte senden Sie mir die ATARI-Computer Fachzeitschrift ab der nächsten erreichbaren Ausgabe ______
für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 80,- frei Haus.
(Ausland: nur gegen Scheck-Voreinsendung DM 100,- Normalpost. DM 130,- Luftpost)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Geschenk- bzw. Lieferadresse:

Name

Vorname

Straße/Nr.

Land PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM ______ liegt bei.

☐ Vorauskasse per Zahlung auf unser Post-scheck-Konto Ffm, BLZ 500 100 60, Kto.-Nr. 5537-602

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberstadt widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Datum Unterschrift

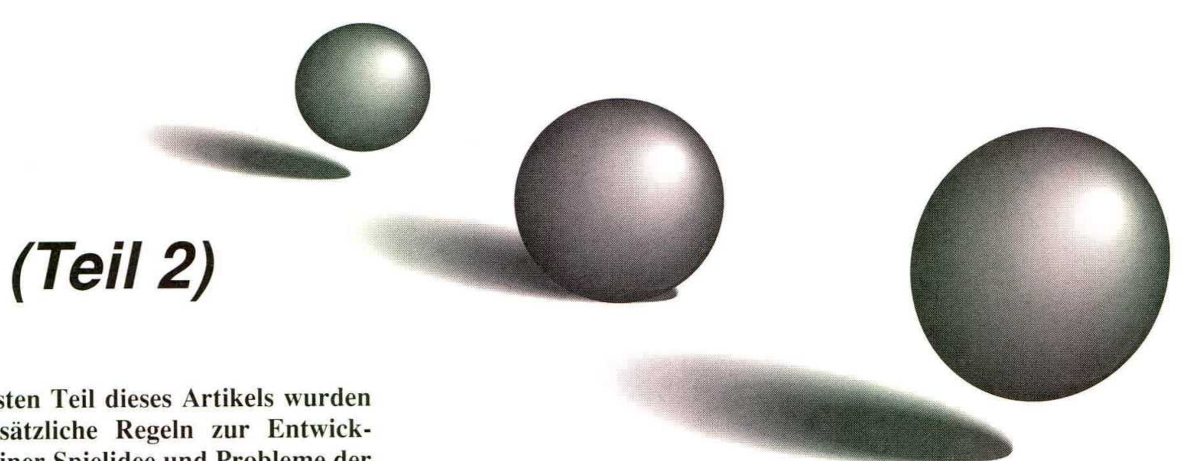
Datum, 2. Unterschrift



ABO

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberstadt widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Professionelles Spiele-Design auf dem ST



(Teil 2)

Im ersten Teil dieses Artikels wurden grundsätzliche Regeln zur Entwicklung einer Spielidee und Probleme der Konzeptionierung beschrieben. Der zweite Teil erläutert weitere Tricks, die bei der Entwicklung des Spiels *esprit* angewendet wurden, und zeigt die Endphase einer Spielentwicklung auf.

Physikalische Effekte

Die Abbildung von physikalischen Gesetzen auf die Welt der Bits und Bytes kann zu ganz neuen Einsichten führen. Was sich in der Natur so leicht nimmt, z.B. die Reflexion eines Balles an der Ecke eines Steines, kann in digitalisierter Form nur sehr unzureichend und mit vielen Tricks nachgeahmt werden. Die Verwendung von komplizierten Differenzialgleichungen ist wegen der Echtzeitdarstellung nicht möglich. Das größte Problem ist dabei die notwendige Quantisierung der Zeit. So darf es z.B. nicht passieren, daß sich bei BOLO der Schläger in dem einem Bild unter dem Ball befindet, und im nächsten Bild aufgrund einer hohen Geschwindigkeit über dem Ball. Bei beiden Bildern würde man ja keine Kollision erkennen können. Trotzdem hätten sich Ball und Schläger durchdrungen.

Was später auf dem Bildschirm sichtbar ist, ist also nicht das, was intern berechnet wird. Die zeitliche und räumliche Auflösung dessen, was berechnet wird, ist wesentlich höher. Deshalb werden die Figurenpositionen durch eine Festkommazahl repräsentiert. Die Definition der Festkommazahl sieht folgendermaßen aus:

```
TYPE FixReal = RECORD
      CASE Zugriff: OF
      TRUE:     I : INTEGER;
      (* Vorkommazahl *)
         F: CARDINAL|
      (* Nachkommazahl in
         1/65336tel-Einheiten *)
      FALSE:    H : LONGINT|
      (* Für den Zugriff
         auf's Ganze *)
      END;
   END;
```

Man erkennt sofort, wie einfach es sich mit solchen Festkommazahlen und der 68000 CPU rechnen läßt. Bei Addition und Subtraktion können die Befehle *add.l* bzw. *sub.l* verwendet werden. Multiplikation und Division sind, je nach gewünschter Genauigkeit, zwar etwas komplizierter, jedoch immer noch schneller als die schnellste Fließkommaberechnung. Zudem ist der Zugriff auf diesen Zahlentyp für die Positionsangabe von Figuren denkbar einfach: Der Vorkommateil entspricht bei einer Positionsangabe der Bildschirmauflösung.

Wie die räumliche Auflösung der Computerwelt, so ist auch die zeitliche Auflösung größer als sie auf dem Bildschirm erscheint. Für jedes Bild, das angezeigt wird, werden alle Bewegungen mit vierfacher Auflösung berechnet. Die Geschwindigkeit einer Figur wird ebenfalls durch eine Festkommazahl dargestellt. Allerdings wird die Geschwindigkeit nicht einfach bei jedem Bildwechsel zur alten Position hinzuaddiert, um die neue einer Figur zu errechnen. Der Geschwindigkeitsvektor wird geviertelt und dann die Figur in vier Zügen mit den jeweiligen Kollisionstests weiterbewegt. Wird eine Kollision erkannt, wird die Figur wieder an die letzte, kollisionsfreie Position zurückgesetzt. Damit wird ein unschönes Druchdringen von Figuren verhindert. Ist eine Kollision passiert, wird eine neue Geschwindigkeit aufgrund des Reflexionsvektors errechnet. Beim nächsten Bildwechsel kann die Figur dann einen (wahrscheinlich) kollisionsfreieren Weg nehmen.

Bei zwei sich bewegenden Figuren kann man sich die Kollisionsberechnung vereinfachen, indem man sich die eine Figur A als bewegungslos denkt und ihre Bewegung als eine relative Bewegung der anderen Figur B ansieht. Weiterhin kann das Begrenzungspolygon der jetzt stehenden Figur A um das Begrenzungspolygon der sich bewegenden Figur B vergrößert werden, so daß man sich die bewegende Figur B als mathematischen Punkt vorstellen kann. Um die Kollision zu berechnen, braucht man nur noch den Bewegungsvektor der Figur B mit dem erweiterten Begrenzungspolygon der Figur A auf Schnittpunkte hin untersuchen.

Interessant ist übrigens, daß die Spielerfiguren, also der Schläger bei BOLO und die Kugel in *esprit*, nicht direkt mit der Maus positioniert werden. Vielmehr wird durch die Anzahl der Mausimpulse pro Bildwechsel eine Kraft repräsentiert, die dann, in Abhängigkeit zur Masse der Spielerfigur, diese beschleunigt (a=F/m). So ist auch die maximale Geschwindigkeit des BOLO-Schlägers durch die träge Masse m und die Bodenreibung r (v'=v*r) begrenzt.

Gleichermaßen genial einfach zu programmieren, wie auch hinterher im Spiel effektvoll, sind die Mulden (Abb.1). Gerät eine Figur auf ein Feld mit einem Muldenobjekt, wird die Figur einfach in Richtung Muldenmittelpunkt beschleunigt. Abb.3 zeigt die Programmierung im Quelltext.

Schattenspiele

Ein sehr beliebter Trick, um ein zweidimensionales Bild plastisch wirken zu lassen, ist die Verwendung von Schatten. Vorzugsweise wird ein Bild von oben links "beleuchtet". Bei den freibeweglichen Figuren in *esprit* wird für diesen Schatteneffekt einfach die Maske der jeweiligen Figur etwas versetzt vorher schwarz gezeichnet. Aufpassen muß man dann allerdings noch auf das Hintergrundbild: Auf dortige Erhebungen darf natürlich kein Schatten fallen! Also werden bei *esprit* alle Steine, auf deren Felder Figuren gezeichnet wurden, vor dem Bildwechsel noch einmal übermalt (siehe auch Teil 1, Abb.2). Gleichzeitig ist somit sichergestellt, daß die Figuren immer unterhalb der Steine dargestellt werden.

Schwieriger wird es bei der Generierung des Hintergrundbildes selbst. Bei den Steinen auf dem Feld ist für die Schattenberechnung mehr Aufwand notwendig (Abb. 2). Jedesmal, wenn ein Stein auf ein Feld gesetzt wird, berechnet sich der Schatten aufgrund der schon umliegend gesetzten Steine. Wenn Sie mit esprit spielen, ist Ihnen das vielleicht noch nicht bewußt aufgefallen: Einige Steine lassen sich ja verschieben und ihr Schatten wird nach dem Verschieben neu berechnet. Oder sind Sie schon auf unterschiedliche Schatten zwischen normalen und schwebenden Steinen aufmerksam geworden? (Dieser Umstand ist in einigen Bildern für die Lösung sehr wichtig!)

Haste Töne?

Den letzten Schliff erhält ein Spielprogramm erst durch den Einsatz von Geräuschen und Musik. (Schalten Sie doch mal den Ton während des Spiels ab!) ATARI hat es einem dabei wirklich schwer gemacht. Eigentlich ist der eingebaute "Knall-&Piepchip" (offizielle Bezeichnung "AY-3-8910") wenig für realistische Geräusche geeignet. Wenn da nicht durch Zufall (??) eine Möglichkeit bestanden hätte, zumindest einen ca. 6-Bit-Digitalsound zu erzeugen, wäre es um die Vertonung von ST-Spielen "ohrenbetäubend" geblieben.

Die relativ einfachen Geräusche bei BOLO wurden durch diskretes Programmieren des Soundchips erzeugt. Bei *esprit* werden nur noch digitalisierte Klänge benutzt. Ein solcher Digitalklang wird mit Hilfe eines ständig laufenden Interrupts erzeugt, der die einzelnen Tondaten, die zuvor mit einem Mikrofon digitalisiert wurden, aus dem Speicher liest und damit die Lautstärkeregister des Soundchips verändert. Dadurch läßt sich der Lautsprecher auf die verschiedenen Positionen "fahren", was in der Abfolge der Daten den eigentlichen Ton erklingen läßt.

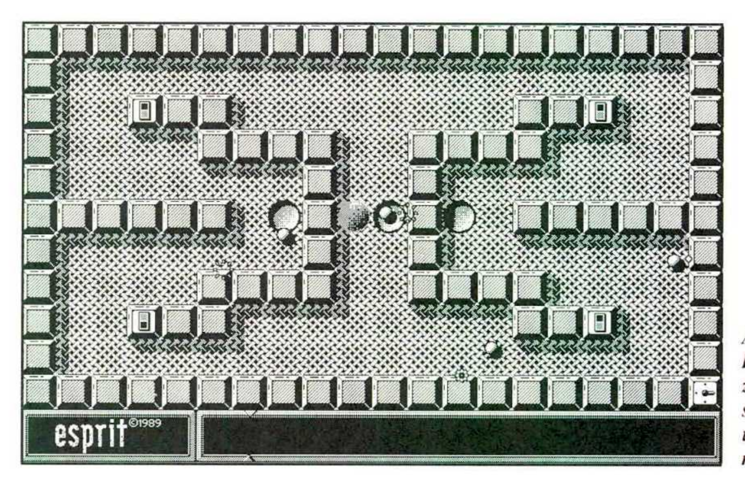


Abb.1: Um alle esprit-Kugeln in die Mulden zu bringen, bedarf es schon etwas Geduld und vor allem einer ruhigen Hand!

Für die Digitalklangausgabe wurde softwaremäßig die Möglichkeit geschaffen, zwei Töne unabhängig voneinander auszugeben. Mehr als zwei Töne hätten zu viel Rechenzeit geschluckt, ein einziger Ton wäre zu unnatürlich gewesen (Sample-Frequenz bei esprit: 6000Hz, Titelmusik 10KHz). Nur zwei unabhängige Töne gleichzeitig ausgeben zu können, erfordert eine gut überlegte Verwaltung der einzelnen Tonkanäle. So müssen die verschiedenen Geräusche verschiedene Prioritäten besitzen, denn manche sind im Spielverlauf wichtiger als andere. Bei längeren Geäuschen ist es auch unangenehm, wenn diese plötzlich von anderen Geräuschen unterbrochen werden.

Die wichtigste Prozedur aus dem Soundmodul ist die Prozedur *StartSound (WhichSound:SOUND; Prioritaet:-INTEGER);. SOUND* ist eine Referenz, die die Sample-Frequenz und einen Zeiger auf die Tondaten enthält. Soll nun in einer Spielsituation ein Ton ausgegeben werden, so wird mit dieser Prozedur der Ton gestartet. Alles andere funktioniert dann automatisch - auch das Abstellen am Tonende. Ein Null-Byte in den Tondaten kennzeichnet das Ende des Klanges. Statt im Ton-Interrupt jedesmal einen Adressenvergleich vorzunehmen, kann man so mit einem einfachen "beq"-Befehl das Ende erkennen. Sollten beim Versuch, einen Ton zu starten, beide Kanäle augenblicklich mit Tönen höherer Priorität besetzt sein, wird der Aufruf ignoriert. Ansonsten wird für den neuen Ton der Kanal mit der niederen Priorität benutzt.

Bei *esprit* müssen natürlich auch bei den verschiedenen Kollisionen entsprechende Geräusche erzeugt werden. Um möglichst allgemein zu entscheiden, bei welcher Kollision welches Geräusch erzeugt werden soll, sind global unterschiedliche Materialien (Glas, Stein, Metall etc.) definiert. Jeder Figur und jedem Steintyp wird ein Material zugeordnet. Über ein zweidimensionales Feld ist dann festgelegt, welches Geräusch beim Zusammenstoß welcher Materialien ausgegeben wird. Über das Material ist übigens auch die Dämpfung des Impuls bei einer physikalischen Kollision definiert.

Noch ein Wort zum vielgelobten DMA-Sound: Um den Sound-Fähigkeiten des Amiga Kontra zu geben, hat ATARI in der Modellreihe STE einen DMA-Sound eingebaut. DMA bedeutet "Direct-Memory-Access", also soviel wie "direkter Speicherzugriff". Die Hardware kann so programmiert werden, daß sie sich, unabhängig vom Prozessor, selbst die Tondaten aus dem Speicher holt und ausgibt. Diese Technik spart dem Programm bei der Ausgabe von Tönen zwar Rechenzeit, macht aber in der Form, wie ATARI die Hardware konstruiert hat, keinen Sinn. Man kann nämlich nur einen Ton gleichzeitig ausgeben. Will man mehrere Töne ausgeben, muß man diese erst mischen und dann wieder im Speicher ablegen. In

der gleichen Zeit kann man aber auch einen Ton per Interrupt abarbeiten. Zudem sind die 8-Bit-DA-Wandler im STE nicht direkt ansprechbar. Dies brachte mich auf den Trick, den DMA-Sound auf einer Adresse stereomäßig repetieren zu lassen und diese dann per Interrupt mit den Tondaten zu beschreiben. Aber NEIN!!! Da die intelligenten Ingenieure bei ATARI aus Geiz das Monochrommonitor-Signal und das DMA-Sound-Ende-Signal "ver-exklusivodert" haben, gab's beim ersten Test auf einem STE nach einigen Sekunden immer wieder einen Kaltstart des gesamten Systems. Das Betriebssystem dachte aufgrund des häufig auftretenden DMA-Sound-Ende-Signals nach einiger Zeit immer, daß der Monochrommonitor abgeklemmt worden wäre. Was der TOS 1.6-Programmierer übersehen hatte, ist, daß der DMA-Sound sich im "Auto-Repeat-Mode" selbst neu starten kann und eben nicht nur unter Programmkontrolle. Da nützt auch das Sperren aller Interrupts während des Monitorsignaltests nichts. Pech!! - soviel zum tollen DMA-Sound des ATARI STE.

Formulierungssache

Das Angebot an Programmiersprachen für den ATARI ST ist vielfältig. Aber welche Spielentwicklung nimmt man? Auf der einen Seite müssen Action-Spiele Geschwindigkeit aufweisen, auf der anderen Seite werden sie in ihrer Ablaufsteuerung sehr komplex. BASIC kann man bei solchen Forderungen gleich in den Mülleimer werfen. Bleiben also noch im engeren Auswahlkreis Sprachen wie Modula, Pascal, C oder Assembler. Die schönste Möglichkeit wäre aber eine Mischung aus einer Hochsprache (Komplexitätsforderung) und einem Assembler (Geschwindigkeitsforderung).

Aus diesem Grunde schreibe ich meine Spiele ausschließlich mit Megamax-Modula-2. Dieses System verfügt über einen Inline-Assembler. Man kann dort an beliebigen Stellen im Modula-Text auf Assembler umschalten, also immer, wenn es zeitkritisch wird, den schnellen Assembler wählen. Die tragende Konstruktion eines Programms stellen die Datenstrukturen dar. Datenstrukturen lassen sich aber am besten in einer Hochsprache definieren. Beim Inline-Assembler des Megamax-Modulas hat man dabei auch den vollen Zugriff auf alle unter Modula-2 definierten Datenstrukturen. Das ist im Gegensatz zu einem normalen Assembler ein enormer Vorteil.

Eines sollte an dieser Stelle einmal klargestellt werden: BOLO und *esprit* wurden mit dem Megamax-Modula-System erstellt. Viele Leute meinen nun, daß hier ausschließlich der Inline-Assembler des Systems benutzt wurde. Dem ist nicht so! Die Verwendung des Inline-Assemblers beschränkt sich nur auf die Moduln, die sehr zeitkritische Aufgaben zu übernehmen haben, so z.B. das Sprite-, das Animations- oder das Sound-Modul. Die globale Steuerung und die Logik der einzelnen Objekte, Steine und Figuren in esprit sind in Modula-2 geschrieben. Würde man das auch in Assembler formulieren wollen, würde man sich heillos in der Komplexität der vielen Ablaufsteuerungen verstricken. Außerdem werden eine ganze Reihe von Datenstrukturen vorher erzeugt und initialisiert, wo es noch nicht so sehr auf extreme Geschwindigkeit ankommt. Und was die Portierung des späteren Programms angeht: Die in Assembler geschriebenen Moduln müßten bei einer Portierung auf einen anderen Rechner sowieso größtenteils - wegen ihren rechnerabhängigen Aufgaben - neu geschrieben werden.

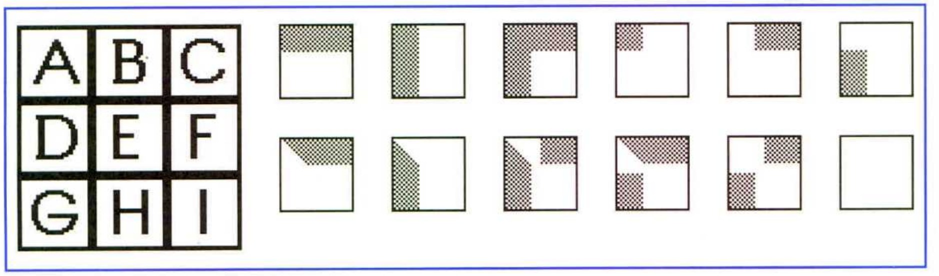


Abb.2: Die Schattenberechnung im Hintergrundbild. Wird ein Stein auf das Feld E gesetzt oder entfernt, ändert sich der Schatten auf den Feldern H,I und F aufgrund der evtl. schon gesetzten Steine auf den Feldern B bis H. Rechts sehen Sie die möglichen Schattenvariationen.

Keine reine Programmiersache

Die Arbeit eines Spieleprogrammierers beschränkt sich nicht nur auf die rein technische Realisation. Neben dieser müssen auch grafische und akustische Gestaltungen erarbeitet werden. Viele, viele Pixel sind hierfür zu setzen. Es hat sich herausgestellt, daß für diese Arbeiten die vielen Funktionen, die die unterschiedlichsten Zeichenprogramme für den ATARI ST bieten, gar nicht benötigt werden. Was man braucht, ist ein einfaches Zeichenprogramm mit einer schönen, schnellen Lupe und einer einfachen Pixel-Setz- und Löschfunktion sowie einer Verschiebemöglichkeit von Bildbereichen, die dann auch noch in der Lupe funktioniert. Funktionen, die nicht innerhalb der Lupefunktion arbeiten, sind einfach nutzlos. Nützlich dagegen sind zwei Grafikseiten, zwischen denen man schnell hin- und herschalten kann. Zum einen kann man so recht einfach Animationen austesten, zum anderen kann auf der einen Seite das Original einer Figur und auf der anderen Seite die Maske einer Figur gemalt werden. Die Maske benötigt man nicht nur wegen des Schattens, vielmehr dient diese zur Begrenzung beim späteren Zeichnen der Figur im Spiel. Wie bei so vielen anderen Tools, muß man sich auch hier sein eigenes erstellen...

Bei der akustischen Gestaltung drängt sich immer die Frage auf: "Wo bekomme ich das Geräusch her?". Nun, entweder man nimmt selbst das Mikrofon in die Hand (was mich - wegen dem Kugelklirren - an den mißglückten Versuch mit dem Hammer und der Cola-Flasche erinnert!), oder man durchwühlt das Angebot einer Videothek. (Ob das Finanzamt wohl die Quittung von der Videothek anerkennt? Das Klirren der esprit-Kugel stammte dann schließlich aus "Die Hexen von Eastwick".) Bei Videos gibt es allerdings eine Schwierigkeit: Die Geräusche, die gebraucht werden, finden sich leicht, doch allzuoft werden diese von anderen überlagert. Eine weitere Möglichkeit an die richtigen Geräusche zu kommen, sind Vertonungs-CDs. Diese sind aber meist sehr teuer und zudem in ihrem Angebot an nutzbaren Geräuschen stark eingeschränkt.

Musik und kleine Jingles erstellt man am besten mit einem Synthesizer. Es ist sehr ratsam, die zu verwendenden Musikstücke selbst zu komponieren. Andernfalls werden Gebühren fällig!

Zur Erstellung der Sound-Dateien benötigt man dann noch einen Sound-Digitalisierer und ein Programm, mit dem man die so digitalisierten Klänge schneiden und zumindest in der Amplitude verändern kann (Abblenden des Tons). Am einfachsten geht man so vor, daß man den Rechner aufnehmen läßt (zwei Minuten gehen bei vier Megabyte bestimmt), dabei die verschiedenen Geräuschversionen einspielt und dann die beste Version digital herausschneidet. Ich benutze zwar

mein eigenes Tool für diese Arbeiten, es gibt aber hierfür auch gute käufliche Programme für den ST (z.B. Sound-Merlin).

Weltenbau

Ein Entwicklungsabschnitt, der besonderen Spaß macht, ist die Erstellung der einzelnen Spielbilder. Endlich sieht man die Früchte seiner Arbeit! Der Aufwand dieser Arbeit kann aber auch leicht unterschätzt werden: Bei esprit waren immerhin einhundert Bilder zu erstellen, wobei viele Bilder wesentlich größer als der Bildschirm sind! Allein diese Arbeiten haben ca. 2 Monate gedauert (also ein Tagespensum von ca. zwei Bildern). Dabei geht es ja nicht darum, nur irgend etwas zu malen, sondern auch darum, sich bei jedem Bild etwas Neues auszudenken. Außerdem waren für viele Bilder neue logische Zusammenhänge, neue Steintypen oder Spezialeffekte zu programmieren.

Testpiloten

Testpiloten müssen bekanntermaßen besondere Fähigkeiten besitzen. Ähnlich ist es bei den ß-Testern eines Spiels. Außer, daß diese Personen sich mit Engelsgeduld bis ins letzte Bild des neuen Spiels vorkämpfen müssen, benötigen sie auch ein besonderes Augenmerk für versteckte Fehler. Aufgrund der Komplexität eines Spiels ergeben sich immer wieder außergewöhnliche Spielsituationen mit katastrophalen Folgen, die es aufzuspüren gilt. Wenn Sie jemanden fragen, ob er ß-Tester für Ihr Spiel werden möchte, werden Sie sicherlich niemanden finden, der dann "Nein!" sagt. Gute ß-Tester muß man sich genau aussuchen! Aber es sind nicht nur Spieleprofis gefragt. Wenn es weniger um das Finden von Fehlern, als um den Test des richtigen Verlaufs der Schwierigkeitskurve geht, holt man sich lieber mal jemanden zum Testspielen, der nur selten ein Computerspielchen wagt.

Diese letzte Entwicklungsstufe ist sehr nervenaufreibend. Nicht nur, weil es um die Eliminierung von ungeahnten Fehlern geht, sondern auch deshalb, weil man dann schon einen Punkt erreicht hat, an dem man sein eigenes Programm nicht mehr sehen mag. Und hier kommt das zum Tragen, was alle guten Programmierer auszeichnet: die Fähigkeit zur Geduld und Ausdauer. Ohne die geht es nicht!

Ein hartes Brot

Wie Sie sehen, ist auch die Spielentwicklung ein hartes Brot mit vielen Problemen und einer langen Entwicklungszeit von der ersten Idee bis zum endgültigen Produkt. Was so locker und leicht anzusehen ist, besteht aus harter Arbeit. Bleibt noch anzumerken, daß sich Spiele - verständlicherweise - nicht so teuer verkaufen lassen, wie andere Anwendungsprogramme, obwohl der Entwicklungsaufwand ähnlich hoch ist (für *esprit*: 8 Monate/ca. 1,3 MByte Text). Umso mehr sind Spiele auf hohe Verkaufszahlen angewiesen. Aber gerade bei Spielen ist der Anteil von Raubkopien besonders hoch. "Es ist doch bloß n' Spiel!" kann man als Argument nicht gelten lassen. Wer seinen Spaß haben möchte, muß auch dafür bezahlen - im Kino genauso wie am Computer.

Meinolf Schneider

```
(* Auszug aus dem esprit-Objekttreiber. Erstellt mit Megamax-Modula-2.
 * (Vertrieb: Application Systems Heidelberg Software GmbH) *)

PROCEDURE DefOBJTrough ( WhichOBJ              : OBJ;
                         pTroughSprite         : Sprite;
                         pTroughRadius         : INTEGER;
                         pTroughEdgeAcc        : Real );
(* Mit dieser Prozedur wird ein neuer Muldengrundtyp
 * definiert. Für die Definition eines Hügels
 * braucht muß man einen negativen Wert bei
 * "pTroughEdgeAcc" angeben! *)
BEGIN
  WITH OBJDefs[WhichOBJ] DO
    OBJTyp := objTrough;
    TroughSprite := pTroughSprite;
    TroughQuadRadius := pTroughRadius * pTroughRadius;
    TroughAccUnit := NegFixReal ( RealToFixReal ( pTroughEdgeAcc /
                                 FLOAT ( LONG ( pTroughRadius ) ) ) );
  END;
END DefOBJTrough;

PROCEDURE FIGCollOBJTrough ( Figure     : FIGPtr;
                             Object     : OBJPtr );
(* Diese Prozedur wird automatisch aufgerufen,
 * wenn sich eine Figur auf einem Feld befindet,
 * die mit einem Muldenobjekt belegt ist. *)
VAR     DistanceVector          : FixRealVector;
BEGIN
  DistanceVector := FieldCenterPos ( Figure^.Position );
  IF QuickVectorLength (Distance) < Object^.TroughQuadRadius
  THEN (* Figur befindet sich in der Mulde: *)
    Figure^.Accelleration := ScaleVector ( DistanceVector,
Object^.TroughAccUnit );
  END;
END FIGCollOBJTrough;
```

Abb.3: So einfach sind die Mulden in esprit implementiert!

Nachtrag

dBMAN Anwendungen

Geschwindigkeit ist keine Hexerei

Es gibt wohl doch mehr dBMAN-Programmierer, als man gemeinhin so denkt. Dies zeigen auch die Zuschriften und telefonischen Anfragen bezüglich der dreiteiligen Serie über das Datenbanksystem dBMAN. Die meisten Schwierigkeiten und Probleme, vor denen die Anwender nicht kapitulieren wollen, sind aber doch so interessant, daß ich es für lohnend gehalten habe, einen vierten Teil anzuhängen.

Keine Angst, diesmal gibt es keine komplette Prozedur. Ich werde aber trotzdem versuchen, die meisten Befehle und Funktionen an einem konkreten Beispiel anschaulich zu machen. Diesmal werde ich auch grundsätzlich auf den Aufbau und die Strukturen von Datenbanken eingehen, da sie ja schließlich das Grundgerüst jeder Anwendung darstellen, und daher die Auswertungs- und Programmiermöglichkeiten von vorneherein beeinflussen, wenn nicht sogar bestimmen.

So beklagten einige Anwender, daß selektierende LIST- bzw. DISPLAY-Operationen bei großen Datenbanken sehr langsam werden. Das ist auch in der Tat so, wenn man nicht ein paar grundlegende Dinge beachtet. Wir wissen ja, daß es die Platten oder Diskettenzugriffe sind, die unseren Programmablauf bremsen. Was die Datenbankstrukturen anbelangt, stellten sich folgende Ursachen heraus:

- Alle Daten werden in eine einzige Datenbank gepackt.
- Es werden unnötige Daten gespeichert.
- Identische Daten werden immer wieder neu gespeichert.
- Wenn mehrere Dateien gleichzeitig geöffnet sind, werden die Relationen zu wenig flexibel gehandhabt.

Übrigens - besonders bei personenbezogenen Daten ist es schon aus Sicht des Datenschutzes höchst bedenklich, alle Daten in einer einzigen Datei zu speichern. Aber auch die Belange der Datensicherung und die Datensicherheit leiden unter zu großen Datenbanken.

Noch etwas, was auch ich mir immer wieder klarmachen muß: Ein Datenbankprogramm geht bei der Suche nach Daten prinzipiell genauso vor wie jeder von uns, wenn er/sie bestimmte Daten aus dem guten alten Karteikasten heraussortieren will. Man/Frau muß jede Karte ansehen und feststellen, ob sie bestimmten Suchkriterien entspricht und ob die gewünschte Information darauf enthalten ist. Anders ist es, wenn ich meine Karteikarten nach eindeutigen Kriterien sortiert habe. Konkret: Sind meine Adressen zum Beispiel nach dem Wohnort sortiert, reduziert sich die Anzahl der Karten, die ich mir ansehen muß, sofort drastisch, wenn ich nach einer bestimmten Adresse in einem bestimmten Ort suche. Sind die Karteikarten innerhalb der Orte zusätzlich noch nach dem Alphabet sortiert, muß ich mir nur noch die Karten eines Ortes mit einem bestimmten Anfangsbuchstaben ansehen. Nichts anderes macht dBMAN, mit dem kleinen Unterschied, daß das Datenbankprogramm nicht wirklich sortierte Daten benötigt, sondern sich mit Indexdateien zufriedengibt.

Wir sehen, alles keine Zauberei - auch eine Datenbank holt nichts aus dem Zylinder, was nicht vorher schon drin gewesen wäre, wenn auch versteckt (verschlüsselt). Bleiben wir im richtigen Leben. Anhand eines Ausschnittes aus den Daten einer Bibliotheksverwaltung möchte ich deutlich machen, wie man oben beschriebene Alltagsweisheit bei der Programmierung und Strukturierung von Datenbanken verwerten kann, und einigen weniger bekannten oder in Vergessenheit geratenen, aber dennoch sehr nützlichen und "powerfull"en Befehlen und Funktionen wieder zu ihrer Daseinsberechtigung verhelfen. Was hätte denn so eine Bibliotheksverwaltung unter anderem zu leisten?

1. Die Daten der Leser und die der Bücher müssen gespeichert und in geeigneter Weise miteinander verbunden werden. Es muß also einerseits möglich sein, ein bestimmtes Buch einem bestimmten Leser zuzuordnen, andererseits müssen die Bücher intern daraufhin ausgewertet werden können, ob bzw. wann sie wieder verfügbar sind.

2. müssen beide Datenbanken bezüglich Rückgabeterminen ausgewertet werden können...also ...wer muß wann welches Buch zurückgeben?

Beziehungen muß man haben...

Wie kann man nun solch eine Datenbank strukturieren und sinnvoll aufbauen? Hier die möglichen Strukturen:

```
FILENAME: LESER.DBF

1 NAME     C  30
2 VNAME    C  20
3 STRASSE  C  30
4 ORT      N   3
5 TEL      C   8
6 SEX      L   1

FILENAME: ORTE.DBF

1 ORT      C  20
2 VORW     C   5

FILENAME: TITEL_1.DBF

1 TITEL    C  50
2 LESERNR  N   4
3 AUSLEIHE D   8

FILENAME: TITEL_2.DBF

1 ISBN     C  13
2 AUTOR    C  20
3 HRSG     C  20
4 VERLAG   C  15
5 GENRE    C  10
6 ....... usw.
```



Ein paar Anmerkungen zur Datei LESER.DBF: Hier sind also die Adreßdaten der Leser. Die weitverbreitete Unsitte, x-mal die Anrede Herr/Frau in der Datenbank zu speichern, war vielleicht zu WORDSTAR's Mailmerge-Zeiten angebracht, wenn man aber ohnehin programmiert, kann man sich pro Datensatz mindestens 3 Bytes sparen, und das läppert sich schon bei 5000 Datensätzen auf satte 14 KB zusammen. Da es ohnehin nur zwei Möglichkeiten gibt (ich hoffe, das diskriminierende "Fräulein" ist inzwischen abgeschafft!), bietet sich ein logisches Feld an - ist doch logisch ! Ich komme später noch einmal darauf zurück.

Die Dateien TITEL_1.DBF und TITEL_2.DBF können über die Datensatznummer direkt verbunden werden, so daß sie unabhängig von einer bestimmten Indexdatei quasi parallel laufen. Will man jedoch nur wissen, welche Leser zu einem bestimmten Zeitpunkt welche Bücher lesen (oder auch nur ausgeliehen haben und nicht lesen), braucht man ja den Ballast von TITEL_2.DBF (Autor, Herausgeber etc.) nicht mitzuschleifen.

Die Datei ORTE.DBF: Ähnlich wie mit der Anrede verhält es sich mit den Wohnorten. In der Praxis wohnen ja meistens mehrere Leute in einem Ort, wieso müssen dann in jedem Datensatz immer wieder derselbe Ort, die gleiche PLZ und Vorwahlnummer geschrieben werden? Verwaltet man die Orte in einer eigenen Datei, muß jeder Ort nur einmal gespeichert werden. In der Datei LESER.DBF verweist dann der numerische Wert im Feld Orte auf die Satznummer des Datensatzes in der Datei ORTE.DBF, wo dann der eigentliche Ort mit PLZ und Ortsvorwahl steht. Ganz nebenbei entsteht so ein Postleitzahlen- und Ortskennzahlenverzeichnis, das als solches schon ganz nützlich sein kann.

Diese vier Dateien werden nun, je nach Bedarf unterschiedlich, verbunden. dBMAN stellt mehrere Möglichkeiten zur Verfügung, zwei oder mehr Dateien miteinander zu verbinden. Einige grundsätzliche Bemerkungen zur Vorgehensweise bei der Definition und der Verwendung von Relationen möchte ich zum besseren Verständnis der Beispiel-Listings an dieser Stelle einfügen.

Zunächst einmal ist die Idee, die hinter der Verbindung von Dateien steckt, die gleiche wie die von Indexdateien, nämlich daß ein Feld einer Datenbank nicht direkt die gewünschte Information enthält, sondern lediglich (in irgendeiner Form) einen Verweis darauf, wo die eigentliche Information zu finden ist. Das kann geschehen durch die Satznummer oder wird repräsentiert durch einen "Schlüssel" (C-String oder numerischer Wert), der grundsätzlich wiederum über einen Index vermittelt werden muß. Hier die beiden möglichen Befehle und deren Syntax:

1. Der dBASE III plus-kompatible Befehl:

```
SET RELATION TO <Ausdruck|Satznummer> INTO <Alias>
```

<Ausdruck> kann wie gesagt sowohl ein C-String als auch ein numerischer Wert sein. Ist beim USE-Befehl kein Aliasname gegeben worden, gilt der Name. Dateiname ohne Extension als <Alias>name.

2. Der flexiblere dBMAN-Befehl

```
SET RELATION <Alias|File ID> TO <Alias|File ID> FIND/GOTO
<Ausdruck> [SCAN | LOOKUP] [SKIP|BLANK|TERMINATE]
```

Der wesentliche Unterschied zwischen beiden Befehlen besteht darin, daß bei dem ersten Befehl die im aktuellen Arbeitsbereich geöffnete Datenbank automatisch zum MASTER wird. Beim zweiten Befehl ist der aktuelle Arbeitsbereich unerheblich. Am anschaulichsten kann das Prinzip wohl an der Verbindung der beiden Dateien LESER.DBF und ORTE.DBF verdeutlicht werden:
In der Datei LESER.DBF steht im Feld ORT wie schon erwähnt nicht der eigentliche Wohnort, sondern die Satznummer RECNO() desjenigen Datensatzes der Datei ORTE.DBF, in dem der Ort, die Postleitzahl und die Telefonortsvorwahl enthalten sind. (Wie sie automatisch dahineinkommt, können Sie dem Listing EINGABE.CMD entnehmen.) Diese Art der Verbindung benötigt keine Indexdatei, da ja durch RECNO() eindeutig ein Datensatz bestimmt wird, auf den dann direkt zugegriffen werden kann. Werden nun beide Daten mit

```
SET RELATION %LESER TO %ORTE GOTO ORT
```

oder, wenn LESER.DBF die aktuell geöffnete Datenbank ist, mit

```
SET RELATION TO ORT INTO ORTE
```

verbunden, setzt dBMAN den Satzzeiger der Datei ORTE.DBF automatisch immer genau auf die Satznummer, die das Feld ORT des jeweiligen Datensatzes der Datei LESER.DBF enthält. Da beim ersten Befehl auch die FILE ID(entifikationen) verwendet werden können, muß bei der Verwendung des Aliasnamens das Aliassymbol "%" vorangestellt werden. Die Felder können nun wiederum mit den Aliasnamen, der FILE-ID (FJ-FS) oder dem Arbeitsbereich (A-J) angesprochen werden. Sind also LESER.DBF in FJ und ORTE.DBF in FK eröffnet, würden folgende drei Zeilen die gleiche Liste mit Name, Vorname, Wohnort, Vorwahl und Telefonnummer liefern.

```
LIST ALL NAME,VNAME,ORTE->ORT,TRIM (ORTE->VWAHL)
                                              +' '+TEL
LIST ALL NAME,VNAME,J.ORT, TRIM(J.VWAHL)+'/'+TEL
LIST ALL NAME,VNAME,B->ORT, TRIM(B->VWAHL)+'/'+TEL
```

Die nächste Möglichkeit, Relationen zu definieren, zeigen die beiden Dateien TITEL_1.DBF und TITEL_2.DBF. Sie werden direkt über die Satznummer verbunden und benötigen deshalb weder einen speziellen Index noch ein gemeinsames Feld.

```
SET REALTION TO RECNO() INTO TITEL_2
```

oder (nicht "und" !) umgekehrt und mit alternativer Syntax

```
SET RELATION %TITEL_2 TO %TITEL_1 GOTO RECNO()
```

Anders ist es, wenn die FIND-Option angewendet wird. In diesem Fall muß ein gemeinsames Feld die beiden Dateien verbinden und die "TO"-Datei nach genau diesem Feld indiziert oder sortiert sein. Jetzt kann auch der Verbindungstyp angegeben werden: LOOKUP bedeutet, daß für jeden Datensatz der Master-Datenbank nur ein einziger Datensatz in der verbundenen Datei erwartet wird. Bei GOTO oder RECNO() ist das ja sowieso klar. Werden für einen Datensatz in der Master-Datei mehrere Daten-



Geschwindigkeit ist keine Hexerei

sätze in der verbundenen Datei erwartet (z.B. Datei ELTERN.DBF mit der Datei KINDER.DBF), so muß die Option "SCAN" angegeben werden. Dies hat zur Folge, daß in der verbundenen Datenbank solange um einen Datensatz weitergeskippt wird, wie der Schlüssel übereinstimmt, dann erst wird der Satzzeiger des Masters auf den nächsten (wahren) Datensatz gesetzt.

Was geschehen soll, wenn für einen Master-Datensatz in der verbundenen Datei kein dem Schlüssel entsprechender Datensatz zu finden ist, wird (auch bei GOTO) durch die dritte Option (Failure Aktion) bestimmt. Wird nichts oder BLANC angegeben, wird in der verbundenen Datei ein leerer Datensatz angezeigt. Die Option SKIP bezieht sich wie oben beschrieben auf den Master, und TERMINATE erzeugt ein EOF(T) der verbundenen Datenbank.

Wie man mit diesen Relationen umgehen kann, wollen wir anhand einiger Aufgabenstellungen unserer Bibliotheksverwaltung durchspielen. Zunächst müssen wir aber alle Datenbanken und Indexdateien öffnen:

```
SELE A
USE LESER INDEX NAME,ORT
SELE B
USE TITEL_1 INDEX AUSLEIHE,LESERNR,TITEL
SELE C
USE TITEL_2 INDEX AUTOR,GENRE...
SELE D
USE D_ORTE INDEX ORTE
```

Auch und gerade bei großen Dateien sollte man sich nicht scheuen, zur Unterstützung von häufig vorkommenden Operationen wenn nötig je eine spezielle Indexdatei anzulegen. Man muß allerdings peinlich darauf achten, daß bei Neueingaben oder Änderungen in den Schlüsselfeldern immer alle Indexdateien geöffnet sein müssen, damit sie automatisch upgedated werden. Hat man's mal vergessen, hilft REINDEX.

Wir wollen nun eine Liste aller Leser mit Name, Adresse und Buchtitel ausdrucken, die ihre Bücher heute zurückbringen müßten, um nicht säumig zu werden. Dazu benutzen wir als MASTER die Datei AUSLEIHE.DBF, verbinden diese über die Lesernummer mit der Datei LESER.DBF und diese wiederum über das Feld Orte mit der Datei ORTE.DBF.

```
SELECT B (TITEL_1)
SET RELATION TO LESERNR INTO LESER
SET RELATION %LESER TO %ORTE GOTO LESER->ORT
```

Nun können wir auf alle Daten, die wir zum Ausdruck benötigen, zugreifen.

Die Verwendung von Indexdateien

Jetzt zur Bedingung: Nehmen wir an, die maximale Ausleihzeit beträgt 3 Wochen. Es sollen also alle Datensätze gedruckt werden, für die gilt:

```
AUSLEIHE = DATE()-3*7
```

Verwendet man nun den SET FILTER TO- oder LIST FOR-Befehl, wird wie in unserem Karteikastenbeispiel die gesamte Datenbank von Anfang bis Ende durchsucht, und das kann bei zwei- oder dreitausend Titeln ganz schön lange dauern, da ja auch unabhängig davon, ob die Bedingung zutrifft, auch die Satzzeiger der verbundenen Dateien weitergesetzt werden.

Wie anfangs schon erwähnt, werden Indexdateien häufig ausschließlich zum Sortieren verwendet. Eigentlich sind sie aber zur Beschleunigung der Suche nach bestimmten Datensätzen gedacht - wie auch Indexregister in Büchern, die ja nicht die Information sortieren, sondern auf Informationen verweisen und sortiert sind, um den Verweis schneller zu finden. Durch das Intervall-Suchverfahren, das ja nur mit sortierten Schlüsselwörtern funktioniert, benötigt man nur n Versuche, um in einer Datenbank mit 2 hoch n Datensätzen einen bestimmten Datensatz zu finden. Man muß also bei ca 15.000 Datensätzen, man lese und staune, nur 14 Datensätze (genauer eigentlich nur 14 Schlüsselwörter) untersuchen, um zum gesuchten Datensatz zu gelangen. Durch diese Erkenntnis ermutigt, indizieren wir also die Datei TITEL_1.DBF nach dem Datumschlüssel:

```
INDEX IN DTOK(TITEL_1->AUSLEIHE) TO AUSLEIHE.
```

Dies hat mehrere nützliche Effekte. Einmal kann jedes gewünschte Ausleihdatum leicht gefunden werden. Gleichzeitig ist die Datei TITEL_1.DBF quasi zweigeteilt, nämlich in zur Zeit nicht ausgeliehene Bücher (deren Ausleihdatum ist ja leer AUSLEIHE={ }) und einem zweiten Teil, in dem am Anfang diejenigen Bücher stehen die am längsten ausgeliehen sind. Und - nicht zu vergessen: sind TITEL_1 und TITEL_2 über RECNO() miteinander verbunden, gilt jeder Index natürlich auch für die jeweils verbundene Datei.

```
SCOPE-Formate

ALL                  alle
ALL FROM <n>         alle ab Satznummer n
RECORD <n>           Satznummer
NEXT <n>             die nächsten n (einschl. des aktu-
ellen)
REST                 den Rest (einschl. des aktuellen)
WHILE <Bedingung>    ... siehe Text

Die wichtigsten dBMAN-BEFEHLE
die den SCOPE-Ausdruck unterstützen

APPEND               Datensätze hinzufügen
AVERAGE              Mittelwert bilden
BROWSE               spaltenweise edieren
COPY                 Datenbank kopieren
COUNT                Datensätze zählen
DELETE               Datensätze löschen
DISPLAY              seitenweise anzeigen
LIST                 Datensätze anzeigen
LOCATE               Datensätze suchen (ohne Index)
MERGE                siehe Text
MODINDEX             siehe Text
RECALL               Löschmarkierungen entfernen
REPLACE              Feldinhalte ersetzen SHOP
Datenbank Menü
SUM                  numerische Feldinhalte addieren
TOTAL                numerische Feldinhalte zusammen
                     fassen und in eine neue Datei
                     schreiben
```

Abb. 1: Besonders bei Befehlen wie SUM oder COUNT ist der Einsatz von WHILE sehr effektiv.



dBMANs vergessene SCOPE-Ausdrücke

Wer schon mal mit dBASE III plus gearbeitet hat, kennt die zu dBMAN unterschiedliche SCOPE- und FOR-Handhabung. Die dBASE-III-Syntax:

```
LIST SCOPE FOR|WHILE <Bedingung>
```

ist bei dBMAN nicht möglich und daher auch bei den Befehlsbeschreibungen im Handbuch nicht zu finden.

Einige Leser, die sich bei mir gemeldet haben, waren der Meinung, daß dBMAN die OPTION WHILE nicht unterstützt, und daher Indexdateien bei LIST und ähnlichen Befehlen nicht oder nur beschränkt benutzt werden können. Weiter vorne im Handbuch findet man aber einen unauffälligen Hinweis, der besagt, daß die WHILE-Option bei dBMAN ein SCOPE-Ausdruck ist. Sie ist damit wesentlich flexibler, nämlich zusammen mit FOR und nicht wie bei dBASE lediglich alternativ zu FOR zu verwenden.

Es ist also auch bei LIST oder DISPLAY dieselbe Prozedur: eine geeignete Indexdatei zum Master-Index machen, den ersten Datensatz mit FIND anspringen und solange (WHILE) listen, wie die Bedingung erfüllt ist.

Für unser konkretes Beispiel hieße das:

```
FIND DTOK(DATE()-3*7)
LIST WHILE AUSLEIH = DATE()-3*7;
  LESER->NAME-' '+LESER->VNAME,TITEL_1->TITEL,ORTE-
>ORT
```

Würde man hier mit LIST FOR AUSLEIHE.. arbeiten, würde der Rest der Datenbank bis zum Ende durchsucht, obwohl die Liste schon längst komplett ist, und gar kein Datensatz mehr gefunden werden kann. Oder - wir wollen alle Titel eines bestimmten Autors auflisten (WHILE), aber natürlich nur diejenigen, die gerade nicht ausgeliehen sind (FOR). Da die Autoren in der Datei TITEL_2.DBF stehen, wird diese zum MASTER und muß nach den Autoren indiziert sein.

```
SELECT TITEL_2
SET INDEX TO AUTOR
FIND Y.AUTOR
LIST WHILE AUTOR=Y.AUTOR FOR TITEL_1->AUSLEIHE = {};
TITEL_1->TITEL,TITEL_2->GENRE
```

Vorher müssen unbedingt alle noch bestehenden Verbindungen wieder gelöscht werden, da sonst unter Umständen Rückkopplungen entstehen könnten, die dBMAN spätestens beim Versuch der Datenausgabe mit zwei Bomben quittiert. Eine Liste der SCOPE-Ausdrücke und der Befehle, die SCOPE unterstützen, finden Sie in Abbildung 1.

Mailmerge mit dBMAN

Natürlich muß unsere Bücherei auch Benachrichtigungen an diejenigen Leser versenden, die ihre Bücher nicht rechtzeitig zurückgebracht haben. dBMAN stellt dazu einen internen MERGE-Befehl zur Verfügung:

```
MERGE <ID> FROM <Dateiname> TO <Dateiname> [SCOPE] [FOR] <Bed> [DELIMETED
<Anf>,<End>]
```

Die Funktionsweise entspricht völlig dem, was man von Serienbrieffunktionen in Textverarbeitungsprogrammen so kennt: Variable Daten werden aus der Datenbank ausgelesen und in einen bestehenden Text an dafür gekennzeichneten Stellen eingefügt - allerdings mit einigen wesentlichen Vorteilen.

1. Das Datenbankprogramm muß nicht verlassen werden.
2. Es muß nicht extra ein ASCII-File erstellt werden.
3. Die Variablen müssen in der Textschablone nicht noch einmal neu definiert werden.
4. Sämtliche dBMAN-FUNKTIONEN sowie alle dem jeweiligen Programm verfügbaren temporären Variablen (Y.X.Z.) können in der Textschablone verwendet werden.

In der Abb.2 können Sie nachvollziehen, wie eine Textschablone aufgebaut sein muß und wie das MERGE-Ergebnis aussieht.

```
MAIL-SCHABLONE MAHN1.TXT
.............................................
Bücherei LESEMUFFEL

                                          <DATE()>

<Y.ANREDE> <J.VNAME-' '+J.NAME>
<J.STRASSE>
<!(M.ORT)>

Sie haben am <K.AUSLEIHE> das Buch <TRIM(K.TITEL)>
ausgeliehen.

Unsere Verleihdauer beträgt maximal 3 Wochen.
Die Rückgabe war also am
<K.AUSLEIHE+3*7> fällig.

...
---------------------------------------------
und ERGEBNIS AUS DER  MAIL-SCHABLONE MAHN1.TXT
.............................................
Bücherei LESEMUFFEL

                                        13.05.1990

Sehr geehrter Herr  Walter Müller
Waldgasse 4                 6236 ESCHBORN

Sie haben am 20.04.1990 das Buch SCHEIBENKLEISTER
ausgeliehen.

Unsere Verleihdauer beträgt maximal 3 Wochen.
Die Rückgabe war also am
11.05.1990 fällig.

...
```

Abb. 2: Eine Schablone für Mailmerge mit dBMAN

Btx/Vtx-Manager

Btx/Vtx: Nase vorn

in der Welt der Telekommunikation mit dem *Btx/Vtx-Manager V3.0.*
Sie wollen Ihr Konto verwalten, Bestellungen aufgeben, eine Urlaubsreise buchen ...
Entdecken Sie jetzt die neuen komfortablen Wege, die Ihnen der *Btx/Vtx-Manager* (als intelligente Komplettlösung) mit dem Abruf aktuellster Informationen und Daten rund um die Uhr liefert.
Ausführliche Informationen erhalten Sie bei Ihrem Atari-Fachhändler oder direkt von uns.

Atari ST Btx/Vtx-Manager V3.0 für 389,– DM *an Postmodem bzw.* 289,– DM *an Akustikkoppler/Hayes-Modem.* (FTZ-Zulassung beantragt). Unverbindliche Preisempfehlungen.

Drews EDV + Btx GmbH
Bergheimerstraße 134 b
D-6900 Heidelberg
Telefon (0 62 21) 2 99 00
Fax (0 62 21) 16 33 23
Btx-Nummer 0622129900
Btx-Leitseite *2 99 00 #





W. Wohlfahrtstätter + J. Ohst EDV GbR

Jutta Ohst * Nelkenstr. 2 * 4053 Jüchen 2
Werner Wohlfahrtstätter * Irenenstr.76 c
4000 D'dorf 30 * ☎ 0211/429876 + 02164/7898







Anschlußfertige Festplatten mit Quantum Laufwerken für ATARI ST

AHS- 40Q	1398 .-
AHS-105Q	1998 .-
AHS-120Q	2398 .-

42,105 bzw. 120 MB Kapazität, 19 ms*)
über 780 kB/s Datendurchsatz
2 Jahre Garantie

AHS-2000	2998 .-

210 MB, unter 20 ms, sehr schnell

AHS-50	1198 .-
AHS-85	1598 .-

mit Seagate Laufwerken,
49.1 bzw. 83.9 MB, 40 bzw. 24 ms *)

AHW-44	1998 .-

mit SyQuest Wechselplatte, incl. Medium
44 MB, 550 KB/s Datendurchsatz, 25 ms *)
Kombinationen mit verschiedenen Platten auf Anfrage.

Der Streamer

AS-155 155 MB, 5 MB/s	1998 .-

Anschlußfertige Laufwerke für ATARI ST

AF-3T	3,5 " Einzelstation 720 kB	228 .-
AF-3TM	3,5 " dto. für "MEGA ST"	228 .-
AF-5	5,25 " 40/80 Track	288 .-

Wir liefern nur getestete Markenlaufwerke von TEAC mit 12 Monaten Garantie

*) Werte beziehen sich auf Angaben der Firmen Seagate, Quantum und SyQuest.

Frank Strauß Elektronik



Schmiedstr. 11, 6750 Kaiserslautern
Tel.: 0631/67096-98 Fax : 60697
Händleranfragen erwünscht





Man muß lediglich beachten, daß dBMAN natürlich alle Serienbriefe in eine Datei schreibt. Will man die Briefe direkt von dBMAN aus drucken, muß am Ende des Serienbriefes ein FORMFEED eingefügt werden. Das ist aber sehr speicherplatzintensiv. Wenn man über eine RAM-Disk verfügt, kann man auch folgendes machen:

```
.....
DO WHILE &Y.BED
DO VAREIN
MERGE FROM M:MAHN.TXT TO M:M_TEMP.TXT
COPY FILE M:M_TEMP.TXT TO PRN:
EJECT
DELE FILE M:M_TEMP.TXT
WAIT
SKIP FJ
ENDDO
....
```

Diese Methode ist besonders interessant, wenn nur einzelne Briefe direkt gedruckt werden sollen, oder wenn man keinen automatischen Einzelblatteinzug besitzt und so Zeit zum Neueinlegen von Papier einplanen muß. Im Progamm-Modul VAREIN.CMD könnten zum Beispiel für jeden Brief noch eigene Anwendereingaben von der Tastatur erfaßt und in den Serienbrief eingefügt werden.

Wie aus der Syntax zu ersehen ist, kann man die Begrenzungszeichen auch selbst definieren, falls die DEFAULT-Begrenzungszeichen "<>" im Text verwendet werden oder man findet, daß die Aliasnamen (LESER->NAME) zur Dateiidentifikation anschaulicher oder übersichtlicher sind. Am sinnvollsten ist dann, einfach '<<' '>>' als Begrenzung zu definieren, dann hat man für den eigentlichen Text weiterhin alle Zeichen zur Verfügung.

```
* PROCDURE EINGABE

* Dies ist ein Beispiel wie man eine Daten -Eingabe /
-Änderungs Prozedur,
* aufbauen kann, die die bequeme Eingabe und Änderung
von Adressen ermöglicht. * Dabei kann der Ort mittels
"Menü" (DISP)ausgwählt werden. * Die Eingabemaske
funktioniert für zwei DBF-Dateien, von denen die eine
* Privat-Adressen (FJ) die andere Institutionen (FK)
enthält mit ent-* sprechend mehr Feldern (Abteilung,
Z.Hd.) .  * Die Datenbank Orte ist sozusagen "selbst-
lernend", da nicht vorhandene * Orte eingegeben werden
können und dann regelmäßig zur Übernahme zur * Verfü-
gung stehen.
ERASE
ASSIGN FRAME(T)
SET GET VIDEO TO 121
SET DELIM OFF
Y.ORT = $(ORTE->ORT,6,9)
X.FID = 'F'+SELECTED()
IF PMENU() = 2 .AND.HMENU() = 1
* Stellt fest ob Neueingabe oder Änderung
Y.ORT = '        '
GO &X.FID 0
* bei Änderung muß der zu ändernde Datensatz gesucht
werden, bevor * in diese Prozedur gesprungen wird.
ENDIF
* Hier wird die Größe der Eingabenaske berechnet.
@  1,1 TO 3,28+VAL($(FIELD(1),AT(',',FIELD(1),2)+1,2))
@  4,1 TO 12,28+VAL($(FIELD(1),AT(',',FIELD(1),2)+1,2))
@  2,02 SAY "EINGABE-MASKE  ADRESSEN  "+ALIAS()
```

Die Sache mit dem SEX

Wie versprochen komme ich nochmal auf das logische Feld SEX in der Datei LESER.DBF zurück. In der Serienbrief-Schablone (Abb.2) können Sie sehen, wie einfach es möglich ist, mit diesem Feld eine geeignete Anrede zu erstellen. Das funktioniert auch in LIST-Befehlen und dem Report-Writer. Selbstverständlich kann man niemanden zumuten, bei der Dateneingabe des Geschlechts Ja oder Nein einzugeben. Mit einem einzeiligen Menü geht das aber sehr elegant:

```
...
* SEX = .T. = 'männl.'
* SEX = .F. = 'weibl.'
Y.SEX = 0
Y.SEX1 = ' weibl'
Y.SEX2 = ' männl.'
@ 1,5 SAY 'GESCHLECHT ' GET SEX SMENU ",",Y.SEX1,Y.SEX2
READ
LESER->SEX = SEX-1
...
```

Das geht, weil ein logisches Feld zwar alle numerischen Wert akzeptiert, aber bei Werten, die größer als eins (>1) sind, immer logisch wahr wird. Y.SEX kann über die SMENU-Funktion nur die Werte 1 oder 2 annehmen, diese werden durch die Subtraktion und durch das Kopieren in das FELD SEX zur logischen 0 oder 1. Aber auch das Filtern von Datenbanken nach Männern und Frauen ist durch das Feld SEX relativ leicht zu bewerkstelligen. Dies geht, wie wir gesehen haben, am besten und schnellsten durch Indizieren der Datenbank. Zwei Dinge sollte man aber bedenken:

1. Wenn man aus bestimmten Gründen solch eine Aufsplittung immer benötigt, liegt es nahe, die Daten in zwei DFB-FILES

```
* 1. Eingabemaske für Privatadressen
IF SELECTED() = 'J' @  5,05 SAY "Name:"
@  6,02 SAY "Vorname:"
@  5,10 GET NAME PICTURE '!XXXXXXXXXXXXXXXXXX'
@  6,10 GET VNAME PICTURE '!XXXXXXXXXXXXXX'
ENDIF
* 2. Eingabemaske für Adressen von Institutionen
IF SELECTED() = 'L' @  5,02 SAY "Institution: "
@  6,02 SAY "Abteilung: "
@  7,02 SAY "zu Händen: "
@  5,16 GET INST PICTURE
'!XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@  6,16 GET ABT PICTURE
'!XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@  7,16 GET ZHD PICTURE
'!XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
ENDIF
* Ab hier sind die Felder wieder in beiden Datenbanken
gleich.
@ ROW()+2,02 SAY "Straße:"
Y.TROW = ROW()+1
@ ROW(),10 GET STRA PICTURE
'!XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ ROW()+1,06 SAY "Ort:" IF PMENU() = 2 .AND.HMENU() =1
@ ROW(),10 GET Y.ORT PICTURE '!XLL'
ELSE
* Bei einer Änderung wird ein anderes Menü ausgegeben,
da die Rufnummer ja
* schon bekann ist @ ROW(),10 SAY ORTE->ORT @ ROW(),40
SAY 'Tel: '+ORTE->VORW+'/'+TEL
@ ROW()+1,10 GET Y.ORT PICTURE "!LLLLLLLL" ENDIF  →
```



Geschwindigkeit ist keine Hexerei

2. Man kann grundsätzlich NICHT nach logischen Feldern indizieren oder sortieren, weil dazu ja der ASCII-Code zugrunde gelegt wird, und der kennt keine logische 0 und 1, die ja gleichzeitig durch .F. und .T. und 'Y' und 'N'. darstellbar sind. dBMAN besitzt aber zwei Funktionen, die uns weiterhelfen.

Übrigens sei an dieser Stelle nochmal darauf hingewiesen, daß sich alle Befehle und deren Syntax auf die Version 5.01 bzw. 5.02 von dBMAN beziehen. Wenn Sie mit einer Version 3.xx arbeiten, sehen Sie erst einmal in Ihrem Handbuch nach, ob der Befehl oder die jeweilige Funktion von Ihrer Version unterstützt werden. Die beiden folgenden Befehle werden Sie vermutlich vergebens suchen.

```
LTON() L(ogik)TON(ummeric)
LTOC() L(ogik)TOC(haracter)
```

Sie erlauben es, einen logischen Ausdruck in eine numerische 0 oder 1 bzw. in ein 'Y' oder 'N' als C-String umzuwandeln. Damit können wir dann auch nach logischen Feldern sortieren. Bei einer Adreßdatei bedient man sich sinnvollerweise der LTOC()-Funktion und hängt den Namen gleich an, so daß gleichzeitig nach dem Alphabet sortiert wird.

```
INDEX ON LTOC(SEX)+NAME TO SEX_N
```

Will man die Indexdatei auch zum Suchen nach Namen verwenden, muß natürlich im FIND-Ausdruck LTOC(SEX) wieder vorangestellt werden:

```
FIND 'N'+Y.VAR
```

oder

```
FIND LTOC(Y.SEX)+Y.VAR
```

Es existiert aber noch eine dritte Möglichkeit: Man kann sich nämlich mit dem Befehl MODINDEX je eine eigene partielle Indexdatei anlegen.

Selektieren mit MODINDEX

Will man also über einen längeren Zeitraum oder immer wieder mit nur einem Teil einer Datenbank arbeiten, ist es manchmal nicht möglich, oder sehr umständlich die Datenbank zu filtern. Hier bietet sich eben die MODINDEX-Methode an. Benutzen wir wieder eine Adressendatei - nehmen wir weiter an, eine Reihe von Operationen soll nur Leser betreffen, die in einem oder mehreren bestimmten Orten wohnen. Prinzipiell kein Problem - was aber, wenn z.B. die "Mobile Bücherei" völlig unregelmäßig verschiedene Orte anfährt, was den registrierten Lesern schriftlich mitgeteilt werden soll? Das heißt, es gibt keinen gemeinsamen Schlüssel, der alle betroffenen Orte erfassen würde. Man kann z.B. mit dem SHOP oder DISPLAY "von Hand" ganz beliebige Orte auswählen und mit dem Befehl

```
MODINDEX [File-ID)] [Bereich] TO [Indexdatei] [FOR] [Bedingung]
ADDITIVE
```

an eine Indexdatei anhängen.

Jetzt aber mal langsam - zum Mitschreiben - denn es sind noch ein paar Vorarbeiten zu leisten, bis das alles funktioniert.

1. Wir legen uns eine LEERE Indexdatei an:

```
INDEX ON ORTE TO ORTE.NDX EMPTY
```

2. Eine möglicherweise noch bestehende Relation zwischen LESER.DBF und ORT.DBF muß erst einmal aufgehoben und in die Datei ORTE.DBF gewechselt werden:

```
READ
* 'Mal CHecKen, ob überhaupt 'was eingegeben wurde..
Y.CHK= $(FIELD(1),1,AT(',',FIELD(1),1)-1)
IF &Y.CHK = '    '
RETURN
ENDIF
SELE FM
* Die Datei ORTE.DBF ist nach Ortsnamen
$(ORT,6,LEN(TRIM(ORT))) indiziert.
IF Y.ORT <> ORTE->ORT
FIND Y.ORT
        ASSIGN WINDOW(13,50,22,78)
IF EOF()
* Falls der Ort noch nicht existiert wird eben ein
N(euer)ORT eingegeben.
DO R_NORT
ENDIF
* Hier kann jetzt der Ort mit dem Cursor angefahren
werden.
DISP NEXT 13 OFF $(ORT,1,26) FOR $(ORT,6,3)=Y.ORT
ERASE
Y.ENTSCH = T
* Man darf sich schon wieder geirrt haben.
@ 1,0 SAY CENTER(ORTE->ORT,28)
@ 2,0 TO 2,28
@ 3,1 SAY 'Übernehmen   Y/N   ' GET Y.ENTSCH PICT 'Y'
READ
IF .NOT. Y.ENTSCH
DO R_NORT
ENDIF
```

```
ASSIGN WINDOW(0,0,24,80)
@ 13,49 CLEAR TO 23,79
@ Y.TROW,06 SAY "Ort: "
@ Y.TROW+1,06 SAY "Tel: "
SET SAY  VIDEO TO 121
@ Y.TROW,11 ESAY  TRIM(ORT)
@ ROW()+1,11 ESAY  TRIM(ORTE->VORW)+'/'
SET SAY  VIDEO TO 7
SELE &X.FID
* Jetzt noch die Rufnummer
@ ROW(),COL() GET TEL PICTURE '############' READ
SELE &X.FID
REPLACE ORT WITH RECNO(FM)
ENDIF
IF PMENU() = 2 .AND.HMENU() = 1
* Letzte Möglichkeit zu Aussteigen.
Y.ENTSCH = Y
@ 22,2 to 24,78
@ 23,5 SAY 'DATENSATZ '+ALIAS()+' NR.:
'+LTRIM(STR(LASTREC(&X.FID)+1,4,0));
+' ANFÜGEN ?'
@ 23,50 SAY '—> ' GET Y.ENTSCH PICT 'Y'
@ 23,58 SAY '<— ' READ
IF Y.ENTSCH
* Jetzt wird der Datensatz angefügt.
APPEND RECORD
ENDIF
ENDIF
RETURN
```

Listing 1



Geschwindigkeit ist keine Hexerei

```
SELE A
SET REALTION TO
SELE D
```

3. Nun kann mit DISPLAY ein beliebiger Ort per Pfeiltasten ausgewählt werden.

4. Zurück in die Datei LESER.DBF. Nun können alle Leser des Ortes, auf dem der Satzzeiger des Arbeitsbereiches D steht, an die (noch leere) Indexdatei ORTE.NDX angehängt werden. Damit der Satzzeiger auch dort bleibt, muß die Relation LESER.DBF->ORTE.DBF aufgehoben werden.

```
MODINDEX TO ORTE FOR ORT = RECNO(FK)
```

oder schneller [wenn LESER.DBF nach Orten (Satznummern von ORTE.DBF) indiziert ist]

```
FIND RECNO(FK)
MODINDEX WHILE ORT = RECNO(FK) TO ORTE
```

Dieser und der vorherige Schritt kann mit weiteren Orten beliebig oft wiederholt werden. Jedesmal werden an die bestehende Indexdatei die der Bedingung entsprechenden Datensätze angehängt.

5. Die Indexdatei wird geöffnet, und die Datenbank enthält lediglich die Leser der ausgewählten Orte.

6. Jetzt kann auch die Verbindung wieder hergestellt werden.

Selbstverständlich müssen die FOR-Bedingungen nicht immer identische sein, so könnten in einem weiteren Durchgang auch noch alle Datensätze angehängt werden, die unabhängig vom Wohnort anderen Kriterien genügen. Beim Filtern mit SET FILTER TO ORT='XY'.OR. ORT=...könnte man schnell an die maximal erlaubte Zeilenlänge geraten.

MODINDEX ist übrigens die einzige "Filter"-Möglichkeit, die auch den Befehl GOTO RECNO() unterstützt und eine Fehlermeldung ausgibt, wenn der anzuspringende Datensatz nicht zum Index gehört. SET FILTER und selbst SET DELETED ON erlauben den Zugriff mit SKIP n oder GOTO n auf einen nicht zum Filter gehörenden oder als gelöscht markierten Datensatz.

Es gibt, wie schon erwähnt, zwei Anwendungsgebiete. Wenn es von der Aufgabenstellung her sinnvoll ist und man verschiedene Teile einer Datenbank öfter getrennt bearbeiten will, legt man sich mehrere solcher "FILTER-Index-Dateien" an und macht sie nach Bedarf zum Master-Index. Dies ist die schnellste Möglichkeit, einen Filter aufzubauen.

Übrigens kann man auch mit dem Befehl UNINDEX einzelne Datensätze - genauer: denjenigen Datensatz, auf dem der Satzzeiger gerade steht, aus der Indexdatei entfernen. Es ist aber auch kein Problem (wenn es sich lohnt), die oben beschriebene Prozedur zu programmieren. (Dies bietet sich besonders dann an, wenn man eine RAM-Disk zur Verfügung hat.) Damit kann sich auch ein dBMAN-unkundiger Endanwender relativ komplexe und extrem schnelle Filter "menügesteuert" selbst zusammenbasteln.

PICTURE - Format & Funktion

Bevor ich noch ein paar Worte zu den PICTURE-Funktionen verliere, möchte zunächst ein paar grundsätzliche Dinge zu STRING-Umwandlungen loswerden. Dies betrifft (an sich) sehr nützliche Funktionen, wie UPPER(), LOWER(), TONAME(),

```
* Dieses Programm erzeugt auf dem Bildschirm einen
übersichtlichen
* Kalender für beliebige also nicht unbedingt
  aufeinanderfolgende Monate.
* 4 beliebige Monate sind gleichzeitig auf dem Bild-
schirm darstellbar * Einfach zusammen mit dem Listing
R_FEI (ST-COMPUTER Jan. '90) abtippen * und in eigene
Anwednungen als "dBMAN-eigenes Accessory" einbauen.

SET TALK OFF
0ERASE
ASSIGN FRAME(0)

Y.CHK = 1
Y.DATUM = DATE()
Y.VARM = MONTH(DATE())
Y.VARJ = YEAR(DATE())

DO R_FEI
* Programmodul zur Berechnung der Feiertage.
* Listing in der Ausgabe Jan. 1990 von ST-COMPUTER

DO WHILE Y.CHK < 5

DO CASE
CASE Y.CHK = 1
        ASSIGN WINDOW(0,0,12,39)
CASE Y.CHK = 2
        ASSIGN WINDOW(0,40,12,79)
CASE Y.CHK = 3
        ASSIGN WINDOW(13,0,23,39)
CASE Y.CHK = 4
        ASSIGN WINDOW(13,40,23,79)
```

```
ENDCASE

@ 4,2       SAY 'Sa'
@ ROW()+1,2 SAY 'So'
@ ROW()+1,2 SAY 'Mo'
@ ROW()+1,2 SAY 'Di'
@ ROW()+1,2 SAY 'Mi'
@ ROW()+1,2 SAY 'Do'
@ ROW()+1,2 SAY 'Fr'
@ 3,0 TO 11, 25 DOUBLE
@ 3,26 TO 11,WCOLMAX()-3
@ 6,8 SAY 'MONAT ' GET Y.VARM PICT '99' RANGE 1,12
@ 7,8 SAY ' JAHR ' GET Y.VARJ PICT '9999' RANGE
1860,2030
READ
IF LASTKEY() = 27
EXIT
ENDIF
IF Y.VARJ <> X.JAHR
DO R_FEI
ENDIF
Y.DATUM =
CTOD('01.'+STR(Y.VARM,2,0)+'.'+STR(Y.VARJ,4,0))
DAT = Y.DATUM @ 6,8 CLEAR TO 7,18
@ 2,2 ESAY &MO +' '+ STR(YEAR(Y.DATUM),4,0)
@ 2,WCOLMAX()-12 SAY 'FEIERTAGE'
Y.DAT = Y.DATUM
Y.COL=3
ASSIGN ROW(DOW(Y.DATUM)+3)
DO WHILE SAMEMONTH(Y.DAT,Y.DATUM)
   Y.COL=Y.COL+3
        DO WHILE ROW() <10 .AND.
SAMEMONTH(Y.DAT,Y.DATUM)          Y.ND0 =Y.DATUM-X.J1
IF NRANGE(ROW(),3,4) .OR. Y.DATUM = DATE() .OR.;    →
```



und die PICTURE-Funktionen mit entsprechender Wirkung. Offensichtlich arbeiten alle diese Funktionen stur nach der Vorschrift: "Wende die Funktion nur auf ASCII-Werte von 65-90 und 97-122 an und addiere bzw. subtrahiere 32, behandle alle anderen Werte als Trennzeichen zwischen zwei Wörtern."

Da alle String-Umwandlungen auf diese Weise über die ASCII-Werte durchgeführt werden, funktionieren sie nicht mit Umlauten und Sonderzeichen, sind also nur eingeschränkt verwendbar. Etwas anderes ist es bei PICTURE-Formatmasken. Eingabemasken wie ...GET Y.VAR PICT '!LLLLLL', erleichtern vor allem Zwei-Finger-Tippern wie mir die Eingabearbeit erheblich, weil man sich schon mal die SHIFT-Taste beim ersten Buchstaben spart.

Um Mißverständnissen vorzubeugen: dBMAN unterstützt PICTURE-Formate und PICTURE-Funktionen. Will man z.B. alle Orte in Großbuchstaben eingegeben haben, so wandelt die Zeile

```
@ ROW(),COL() SAY 'ORT ' GET Y.ORT PICT '@!'
```

die Umlaute nicht um, denn hier handelt es sich um eine PICTURE-Funktion, die mit dem 'Klammeraffen' eingeleitet wird. Eine PICTURE-Maske hingegen:

```
@ ROW(),COL() SAY 'ORT: ' GET Y.ORT PICT '!!!!!!!!!!!!'
```

wandelt auch die Umlaute (nicht das "scharfe S") korrekt um. Dennoch gibt es mindestens zwei PICTURE-Funktionen, die ganz sinnvoll eingesetzt werden können.

Das europäische DATUM-Format
...

```
Y.DATUM = {}
@ ROW(),COL() SAY 'DATUM ' GET Y.DATUM
PICT '@E'
READ
....
```

Einige Leser haben auch gefragt, was eigentlich die Picture-Anweisung im Zusammenhang mit "SAY ..." für einen Sinn hat. Nun - wieder mal - Speicherplatz sparen !. Es ist manchmal sinnvoll, "Trennzeichen", die immer identisch sind, nicht in der Datenbank zu speichern, sondern eben nur die puren Daten. Nehmen wir als Beispiel die Bankleitzahlen. Sie erscheinen grundsätzlich in der Form '### ### ##'. Es ist doch nicht nötig, die beiden Leerzeichen (2 Bytes) ständig mitzuschleifen. Die PICTURE-Funktion "@R" (man kann sich merken: "R" wie "R"eduziert) bewirkt, daß Formatzeichen, wie in unserem Fall eben die Leerzeichen, nicht in der jeweiligen Variablen gespeichert werden. Folgende Syntax ist zu beachten:

```
...
Y.BLZ = SPACE(8)
3,5 SAY 'BLZ' GET Y.BLZ PICTURE '@R ### ### ##'
READ
? Y.BLZ

ergibt z.B. '1002ßß00'
aber

? TRANSFORM(Y.BLZ,'### ### ##'

ergibt wieder '100 200 00'
```

Ja - genau - mit der TRANSFORM-Funktion (! Achtung Umlaute !) lassen sich die PICTURE-Masken auch in LIST- und DISPLAY-Befehlen verwenden. Am Beispiel unserer Dateien LESER.DBF und ORTE.DBF könnte man eine Liste der Telefonnummern folgendermaßen ausgeben.

```
LIST ALL NAME-' '+VNAME+ 'TEL.: ' +;
    TRANSFORM(ORTE->VWAHL+LESER->TEL,'#####/#######)
```

Da die Telefonnummer aus der Rufnummer und der Vorwahl zusammengesetzt ist, ist ja das Trennzeichen "/" noch nicht enthalten und kann so mit TRANSFORM bequem immer an der richtigen Stelle eingefügt werden.

```
NLIST(Y.ND0,X.ND1,X.ND2,X.ND3,X.ND4,X.ND5,X.ND6,X.ND7,
     X.ND8, X.ND9,X.ND10,X.ND11,X.ND12,X.ND13,X.ND14,
                                          X.ND15)<>0
Y.NLISTE = ;
NLIST(Y.ND0,X.ND1,X.ND2,X.ND3,X.ND4,X.ND5,X.ND6,
              X.ND7,X.ND8,X.ND9,X.ND10,X.ND11,X.ND12,
                             X.ND13,X.ND14,X.ND15)
        IF Y.NLISTE <>0
           Y.RW = ROW()
           Y.CL = COL()
           SET SAY VIDEO TO 7
           @ ROW()+1,WCOLMAX()-12 SAY EXTRACT('/
',X.FLISTE,Y.NLISTE)
           ASSIGN ROW(Y.RW),COL(Y.CL)
        ENDIF
        SET SAY VIDEO TO 112
      ELSE
        SET SAY VIDEO TO 7
      ENDIF
           @ ROW()+1,Y.COL SAY STR(DAY(Y.DATUM),2,0)
           Y.DATUM = Y.DATUM+1
        ENDDO
  ASSIGN ROW(3)
ENDDO
SET SAY VIDEO TO 7
@ 22,1 SAY ' '
INC Y.CHK,Y.VARM
ENDDO
WAIT
ASSIGN WINDOW(0,0,24,80)
ERASE
RETURN
```

Listing 2

```
ERASE
SELE A
ANF=' '
END=' '
* Eingabe der Bandbreite
@ 1,5 GET ANF
@ 2,5 GET END
READ
FIND ANF
ERASE
In der RANGE -Option werden Anfang und Ende einfach
als Makros eingesetzt.
DISP OFF WHILE NAME like '[&ANF-&END]*' NAME-'
'+VNAME,D_ORTE->ORT
ERASE
RETURN
```

Listing 3



Last aber .NOT. least, gibt es noch eine PICTURE-Funktion für MEMO-Felder. Also angenommen, in der Datenbank TITEL_2 existiert noch ein MEMO-Feld namens TITEL_2->INHALT, das eine kurze Inhaltsbeschreibung des jeweiligen Buches enthält. Der Befehl

```
@ 10,10 SAY TITEL_2->INHALT PICT '60,8'
```

bewirkt die Ausgabe der Inhaltsbeschreibung in der Zeile 13, Spalte 10, und zwar mit einer Breite von 60 Spalten und einer Länge von 8 Zeilen, mit WORD-WRAP und allem Drum und Dran. Man könnte auch sagen, der Text erscheint in einem Bildschirmfenster mit den Koordinaten 10,10,18,70. Die Umschaltung mit SET MEMOWITH TO kann man sich ersparen, dieser SET-Befehl trägt wahrscheinlich nur der dBASE III plus-Kompatibilität Rechnung.

String-Suchoperationen

Verlassen wir unsere fiktive Bibliothek und wenden wir uns konkreten Büchern zu. Zugegeben, das englische Handbuch von dBMAN animiert nicht gerade zum Lesen. Über den Preis für das deutsche "UPDATE" schweigt der Gentleman und denkt sich seinen Teil. Manche Kapitel werden wohl aus diesem Grund von den meisten mit dem Hinweis "kenn' ich schon" übersprungen. Ich habe den Eindruck, daß das auch auf den Abschnitt über die String-Suchoperatoren zutrifft.

```
* PROCEDURE L_ORTE

* Mit diesem kleinen Testprogramm können sehr schnell
und bequem * alle Adressen oder Telefonnummern eines
bestimmten Ortes * aufgelistet werden
*
* CLOSE ALL
* SELE FM
* USE ORTE INDEX ORTE * SELE FJ
* USE LESER INDEX ORTE1

SELE FJ
ERASE
* SCHLÜSSEL ORTE1.NDX = TRIM($(ORT,5,LEN(TRIM(ORT))
* SCHLÜSSEL  ORTE.NDX = STR(ORT,3,0)+TRIM(NAME)
* ORT SUCHEN
Y.ORT = '      '
@ 1,5 SAY 'WELCHER ORT' GET Y.ORT PICT '!LLLL'
READ
ERASE
FIND FM TRIM(Y.ORT)
IF .NOT. EOF(FM)
    SELE FM
    * Auswahl des Ortes
    DISP OFF WHILE $(M.ORT,6,LEN(TRIM(Y.ORT))) =
TRIM(Y.ORT) M.ORT
    FIND FJ STR(RECNO(FM),3,0)
    IF FOUND(FJ)
       ERASE
       SELE FJ
       * Kopfzeile
       ? 'WOHNORT '+D_ORTE->ORT,'VORWAHL:'+
                                         D_ORTE->VORW
       ?
       LIST OFF WHILE J.ORT = RECNO(FM) !(NAME)-'
'+VNAME,'TEL.: '+TEL          WAIT      ENDIF
ENDIF
ERASE RETURN
```

Listing 4

Außer <var>$"<var> stellt dBMAN noch <var> LIKE <var> zur Verfügung, und dieser Operator erlaubt die Option RANGE.

Die genaue Syntax und die Anwendungsmöglichkeiten mache ich am besten an einem Beispiel für eine Benutzereingabenüberprüfung deutlich. In vielen Anwendungen kommt es vor, daß Ein- oder Mehrbuchstaben-Codes für Steuerklassen, Warenkategorien oder ähnliches eingegeben werden müssen. Diese umfassen aber meist nicht das ganze Alphabet. Nehmen wir an, als CODE sind die Buchstaben A-J (z.B. für FILE IDs) erlaubt. Jede andere Eingabe soll verweigert werden.

```
Y.FID = ' '
@ ROW(),COL() SAY 'CODE: ' GET Y.CODE PICT '!' VALID .....
```

Die Überprüfung der Eingabe erfolgt mit VALID. So weit, so gut. Viele Programmierer lösen das so:

```
...VALID Y.CODE >= 'A' .AND. Y.CODE <= 'J'
```

Mit LIKE und RANGE geht es um einiges einfacher.

```
... VALID Y.CODE LIKE '[A-J]'
```

Im Listing L_NAME.CMD können Sie sehen, daß diese Möglichkeit auch mit Makros und Platzhaltern funktioniert! RANGE muß immer in eckigen Klammern eingeschlossen sein, mit "-" wird der Bereich angegeben. Auch folgendes ist erlaubt:

```
Y.VAR = 'B'
? Y.VAR LIKE '[AC-F]'
. F
```

Diese Zeile liefert F(alsch), denn '[AC-F]' bedeutet, daß die Buchstaben A,C,D,E und F erlaubt sind, nicht jedoch B. Wer will sowas denn noch mit .AND., .NOT. und .OR. programmieren? ... Man sieht also, es gibt eine Menge Gelegenheiten, diese Option einzusetzen, denn zusätzlich können auch noch Platzhalter ("?" ; "*") an beliebiger Stelle eingesetzt werden. Es soll auch nicht unerwähnt bleiben, daß <var>$<var> auch für MEMO-Felder funktioniert. So kann, wenn in MEMO-Felder z.B. Produktbeschreibungen stehen, nach beliebigen Schlagwörtern oder Eigenschaften, die in den Beschreibungen enthalten sein sollen, gesucht werden.

Ein eidgenössisches Rundungsproblem

Das die Eidgenossen ein besonderes Verhältnis zum Geld haben, ist hinlänglich bekannt. Jedenfalls habe ich durch Zuschriften aus diesem Land, das mir als Allgäuer von der Mentalität her sehr nahe steht, gelernt, daß man sich dort mit Kleinkram - sprich Rappenbeträgen, nicht mehr abgibt. Es werden nämlich grundsätzlich die Hunderstelstellen auf "5" gerundet. Diese Funktion ist natürlich in dBMAN nicht implementiert. Listing 3 ist ein Testprogramm, das zeigt, wie man derlei Probleme kurz, aber mitunter keineswegs schmerzlos, lösen kann. Wie es genau funktioniert, werden Sie sicher selbst herausfinden. Hier das Prinzip, das sich

Nachtrag

dBMAN Anwendungen

Geschwindigkeit ist keine Hexerei

auch für andere Problemlösungen verwenden läßt, und nochmal Gelegheit bietet, sich mit den IIF-Funktionen zu befassen.

Beim Problem selbst geht es ja im wesentlichen darum, festzustellen, ob überhaupt eine 5er-Rundung vorgenommen werden muß. Extra gerundet werden muß ja nur bei allen Hunderstelwerten von 3;4 und 6;7, bei allen anderen Werten kann ja mit der dBMAN-eigenen ROUND()-Funktion gearbeitet werden. Von dieser ersten "groben" Zehntelrundung gehen nun, falls nicht bereits eine '0' oder'5' an der zweiten Stelle nach dem Komma stehen, alle weiteren Berechnungen aus. Jetzt gibt es wieder zwei Unterscheidungen: Es müssen nämlich nur zu diesem mit ROUND(n,1) auf Zehntel gerundeten Wert 0.05 addiert oder subtrahiert werden. Der Ausdruck:

```
ROUND((5-MOD(FRAC(Y.ZAHL)*100,10))/2,0)*0.05
```

ändert das Vorzeichen von 0.05, je nach Bedarf, so daß Addition bzw. Subtraktion ohne eigene IF-Anweisung erfolgen können. Na ja ... es bleiben ja nur noch die Werte '3' (*1/100) und '7' (*1/100) übrig. Der Betrag der Differenz von 5-3 bzw. 5-7 ist jeweils gleich, nämlich '2'. Das heißt, sie unterscheidet sich nur durch das Vorzeichen. Dividiert man dieses Ergebnis durch 2, erhält man je nachdem (+1) oder (-1). Durch die Multiplikation wird das Vorzeichen von '0.05' immer richtig angepaßt.

Ich denke, mit diesen Hinweisen ist es kein Problem mehr, die vier Zeilen des Listings, die die eigentliche Rundung betreffen, nachvollziehen zu können.

Die PRINTSTAT()-Bomben

Leider ist es richtig - die Funktion PRINTSTAT(n), die eigentlich den ON-/OFF-LINE-Zustand des Druckers übergeben soll, liefert regelmäßig zwei Bomben, wenn ein Drucker angeschlossen ist. Ich weiß weder, woran es liegt, noch wie es zu beheben ist. Einen kleinen Trick, den mir ein Leser übermittelt hat, will ich Ihnen nicht vorenthalten. Es bleibt ja erst mal nichts anderes übrig, als PRINTSTAT() nicht zu verwenden. Trotzdem kommt es immer wieder vor, daß der Drucker noch OFF-LINE steht. dBMAN macht dann wahrscheinlich folgendes: Es wird etwa 30 Sekunden lang versucht, die Daten über die angegebene Schnittstelle loszuwerden; wenn das nicht gelingt, werden sie in das Daten-Nirwana - sprich: DEVICE = NULL: - sprich: in den Papierkorb geschickt, und dBMAN fährt im Programm fort. Da die meisten Drucker über einen Puffer verfügen, der die gesamten Daten in ein oder zwei Schüben aufnehmen kann, geht die Übermittlung in Sekundenbruchteilen vor sich, auch wenn der Druck selbst noch längst nicht beendet ist. Der Trick ist nun, einfach mittles TIME() zu Beginn des Druckvorgangs, also am besten schon bei der Übermittlung der Steuersequenzen, die Zeit zu speichern und mit der Zeit zu vergleichen, die am Ende des Druckvorgangs, oder richtiger am Ende der Datenübertragung, genommen wird. Ist nun mehr als eine halbe Minute vergangen, kann man davon ausgehen, daß der Drucker nicht eingeschaltet war, und eine entsprechende Warnung ausgeben lassen.

Peter Neuchel

Programmer's ´Toolbox - Dateien

Teil 2: Die GEMDOS-Betriebssystemfunktionen

Im Zentrum des ersten Blocks der "Programmer's Toolbox" stehen die Betriebssystemfunktionen des ST und ihre Anwendung durch die Kommandos der Shell. Da dabei "gut" die Hälfte aller GEMDOS-Funktionen Anwendung findet, erscheint es sinnvoll zunächst einen kurzen Ausflug in diesen Bereich des ST-Betriebssystems durchzuführen, um die nachfolgenden Serienteile nicht übermäßig stark durch diesbezügliche Erläuterungen zu belasten.

Einen guten Einstieg in die Thematik erhält man mit Hilfe von Abb.1.4. Hier dargestellt finden Sie die einzelnen Komponenten des ST-Betriebssystems. Im wesentlichen liegt eine Differenzierung zwischen TOS (Tramiel? Operating System) und GEM (Graphics Environment Manager) vor. Das GEM beschäftigt sich mit der Verwaltung der (höheren) grafischen Funktionen des ST und ist damit für uns von untergeordneter Bedeutung, da innerhalb dieser Artikelserie ausschließlich auf der Textebene gearbeitet wird. Das TOS dagegen bildet die Schnittstelle zur Hardware des ST, daher sind auch in diesem Bereich die von uns benötigten Funktionen angesiedelt und es sollen zunächst die Komponenten des TOS näher betrachtet werden.

TOS setzt sich aus drei Komponenten zusammen, in denen zwischen geräteunabhängigen und geräteabhängigen Funktionen unterschieden wird. Die geräteunabhängigen und damit die höheren Funktionen des TOS sind im GEMDOS (Graphics Environment Manager Disc Operating System) untergebracht. Die geräteabhängigen Funktionen sind im BIOS (Basic Input Output System) und im XBIOS (eXten-ded Basic Input Output System) angesiedelt. Das BIOS enthält dabei sämtliche Funktionen, die zur Implementierung des GEMDOS benötigt werden. Das XBIOS enthält weitere Funktionen, die besondere Eigenschaften der ST-Hardware unterstützen und nicht von GEMDOS verwendet werden.

Bei den Arbeiten zur "Programmer's Toolbox" hat sich herausgestellt, daß die im wesentlichen auf den Bereich des Dateisystems fixierten Funktionen des GEMDOS für die gewünschten Implementierungen ausreichend sind. Im folgenden findet sich daher ein knappes, aber vollständiges Kurz-Reference für sämtliche GEMDOS-Funktionen. Es ist zu beachten, daß die Funktionen nicht in der relativ willkürlichen Reihenfolge aufgeführt sind, in der sie im Betriebssystem implementiert sind. Stattdessen habe ich sie in funktionalen Gruppen zusammengefaßt1. Generell gilt, daß bei Anwendung der Funktionen immer die Header-Datei "OSBIND.H" verwendet werden muß, da in ihr die entsprechenden Deklarationen vorgenommen sind. Ferner gilt, daß negative Funktionsergebnisse bei allen Funktionen ein Fehlverhalten signalisieren, das entsprechend Abb.1.5 zu interpretieren ist. In den nachfolgenden Funktionsbeschreibung wird daher auf eine separierte Dokumentation von Fehlverhalten verzichtet.

Ein- und Ausgabe mit der Konsole

ANWENDUNG

```
chr = Cconin();
chr = Crawcin();
chr = Cnecin();
Cconout(chr);
chr = Crawio(wrd);
status = Cconis();
status = Cconos();
Cconws(str);
Cconrs(str);

short status, chr, wrd;
char *str;
```

BESCHREIBUNG

Cconin, ***Crawcin*** und ***Cnecin*** holen jeweils das nächste Zeichen von der Konsole. *Cconin* arbeitet dabei mit Echo. Die beiden anderen Funktionen arbeiten ohne Echo. *Cconin* und *Cnecin* interpretieren die drei Kontrollzeichen:

^S Ausgabe stoppen
^Q Ausgabe fortsetzen
^C Programm abbrechen

Crawcin interpretiert diese drei Zeichen nicht. Es liefert stattdessen ihren ASCII-Wert.

Cconout gibt das Zeichen *chr* auf der Konsole aus.

und Ausgabe. Wenn *wrd* ungleich 0xFF ist, wird *wrd* ausgegeben, ansonsten wird *chr* gelesen.

Cconis gibt einen Wert ungleich Null zurück, wenn ein Zeichen an der Konsole bereitliegt.

Cconos gibt einen Wert ungleich Null zurück, wenn die Konsole bereit ist, ein Zeichen zu empfangen.

Cconws schreibt einen nullterminierten String auf die Konsole.

Cconrs liest einen nullterminierten String der Maximallänge *str[0]* in den Vektor ab Position *str[2]* ein. *In str[1]* wird dabei die aktuelle Länge des eingelesenen Strings abgelegt.

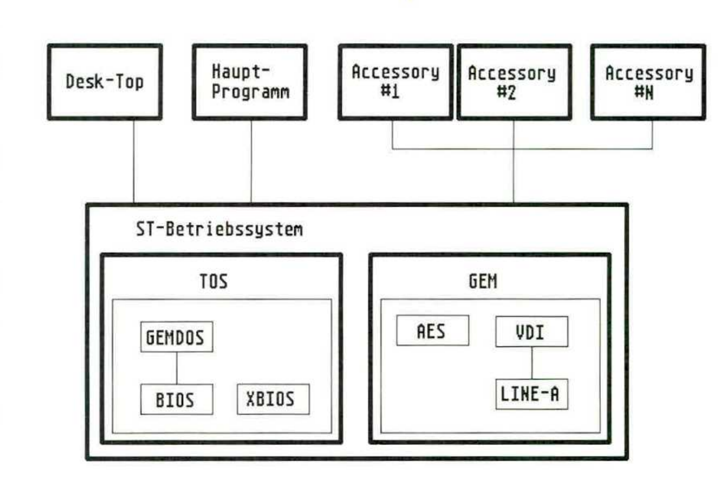


Abb. 1.4: Architektur von Desk-Top und TOS

Nr.	Bedeutung
-1	allgemeiner Fehler
-2	angesprochenes Gerät nicht bereit
-3	unbekannter Befehl
-4	CRC-Fehler
-5	ein gegebener Befehl ist ungültig
-6	ein angesprochener Track kann nicht gefunden werden
-7	ungültiger Bootsektor
-8	ein angesprochener Sektor kann nicht gefunden werden
-9	Drucker nicht bereit
-10	Schreibfehler
-11	Lesefehler
-12	allgemeiner Fehler
-13	Diskette schreibgeschützt
-14	Diskette wurde gewechselt
-15	unbekanntes Gerät
-16	beim Verifizieren wurden kaputte Sektoren gefunden
-17	Diskette einlegen (bei einem Laufwerk)
-32	ungültige Funktionsnummer
-33	Datei nicht gefunden
-34	Pfadname nicht gefunden
-35	zuviele offene Dateien
-36	Zugriff nicht möglich
-37	ungültige Handle-Nummer
-39	ungenügender Speicher
-40	ungültige Speicherblockadresse
-46	ungültige Laufwerksbezeichnung
-49	keine weiteren Dateien

Abb.1.5: Fehlermeldungen von GEMDOS, BIOS und XBIOS

Ein- / Ausgabe mit der seriellen Schnittstelle

ANWENDUNG

chr = Cauxin();
Cauxout(chr);
status = Cauxis();
status = Cauxos();

short chr, status;

BESCHREIBUNG

Identisch mit *Cconin, Cconout, Cconis* und *Cconos*, mit dem Unterschied, daß die Ein- und Ausgaben auf der seriellen Schnittstelle anstelle der Konsole erfolgen.

Ausgabe auf die parallele Schnittstelle

ANWENDUNG

status = Cprnout(chr);
status = Cprnos();

short chr, status;

BESCHREIBUNG

Identisch mit *Cconout* und *Cconos*, mit dem Unterschied, daß die Ausgaben auf der parallelen Schnittstelle anstelle der Konsole erfolgen.

Bearbeitung von Geräten

ANWENDUNG

drv = Dgetdrv;
map = Dsetdrv(drv);
Dfree(buf, drv);

long map;
short drv;
long buf[4];

BESCHREIBUNG

Dgetdrv ermittelt die Nummer des aktuellen Geräts. Dabei steht der Wert 0 für Gerät A, der Wert 1 für Gerät B und so weiter.

Dsetdrv setzt das aktuelle Gerät. Der Parameter *drv* kodiert das Gerät dabei auf genau die gleiche Weise wie die Ergebnisse von *Dgetdrv* zu interpretieren sind. Als Ergebnis (*map*) von *Dgetdrv* erhält man einen Bit-Vektor, der alle angeschlossenen Geräte anzeigt. Ein gesetztes Bit 0 zeigt dabei ein existierendes Gerät A an, ein gesetztes Bit 1 steht für Gerät B und so weiter.

Mit ***Dfree*** können Geräteinformationen abgefragt werden. In *drv* ist dabei auf die oben gezeigte Weise das Gerät zu kodieren. Als Ergebnis erhält man einen Vektor von vier *long*-Werten mit folgender Bedeutung:



POWER PER POST

SEMAG TS neuen eid treitnesärp



KLAX

Die Umsetzung des brandheißen Automatenspieles! Sammeln Sie Steine, die Ihnen auf einem Förderband entgegenkommen, und legen Sie sie in die richtige Fächer. Kein Problem - oder doch? Denn für die neun unterschiedlichen Steinen gibt es nur 5 Fächer. Suchteffekt á la Tetris ist garantiert, ein 2-Spieler Modus (gleichzeitig!) ist vorhanden.

Best.-Nr. 80 DM 59,90

Ultima V

Das hervorragende Fantasy-Rollenspiel jetzt auf Ihrem ST. Die Odysee durch Britannia geht weiter. Finden Sie Freunde, die Ihnen bei der Suche nach dem verschollenen Lord British behilflich sind.



Best.-Nr. 923 DM 79,-

An alle ST User

Mehr Informationen (auch über PD) finden Sie in unserem Hauptkatalog.

Sierra total

Goldrush	65,90
King's Quest 1-3	84,90
King's Quest IV	84,90
Larry 1	67,90
Larry 2	72,90
Manhunter 1	72,90
Manhunter 2	79,90
Police Quest 1	67,90
Police Quest 2	65,90
Space Quest 2	67,90
Space Quest 3	72,90

Wir bieten auch zu diesen Sierra-Spielen die Lösungswege an. Für nur DM 12,- je Lösung oder alle komplett im Ringbuchordner für DM 79,-



Aktuell

Pipemania	59,90
W. Gretzky Hockey	69,90
Rock Star	24,90
Midwinter	79,90
Starflight	74,90
Xenomorph	69,90

PD

In unserem Hauptkatalog haben wir eine große Anzahl an guten und ausgesuchten PD-Programmen.



07252/3058

Bestellen Sie noch heute Ihr Game

Lieferung per Vorauskasse, zuzügl. DM 4,- (Ausl.6,-), Nachnahme DM 6,50 (Ausl.10,-)

Fordern Sie unseren kostenlosen Gesamtkatalog an !

☐ **ST-Katalog**

Name

Vorname

Straße

PLZ/Ort

Coupon ausschneiden und senden an:

Power per Post (Inh. W.Rätz)
Postfach 1640
7518 Bretten

STC 07/90

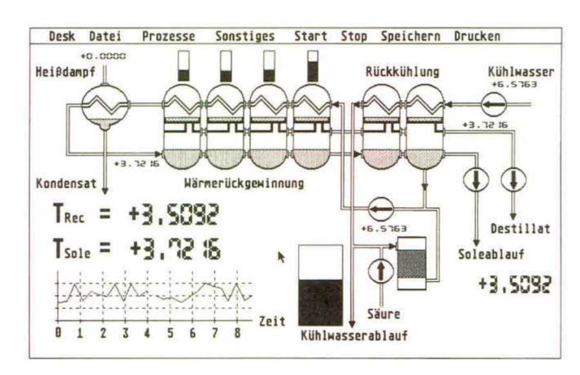
Entenmühlstraße 57
6650 Homburg/Saar
Telefon (06841) 64067
Telefax (06841) 2467





Anwendung des Monats:

Chemieanlagen-überwachung



Komplettlösungen mit den Hard- und Softwareprodukten aus dem Hause rhothron GmbH

Computer Designed Instrumentation

buf[0]	Anzahl der freien Cluster eines Geräts
buf[1]	Gesamtzahl der Cluster
buf[2]	Anzahl der Bytes pro Sektor
buf[3]	Anzahl der Sektoren pro Cluster

Bearbeitung von Dateien

ANWENDUNG

handle = Fcreate(name, attr);
error = Fdelete(name);
handle = Fopen(name, mode);
error = Fdup(stdhandle);
error = Fforce(stdhandle, handle);
error = Fclose(handle);
rc = Fread(handle, count, buf);
rc = Fwrite(handle, count, buf);
pos = Fseek(offset, handle, mode);

char *name, *buf;
short handle, stdhandle, error, attr, mode;
long rc, count, pos, offset;

BESCHREIBUNG

Fcreate erzeugt eine Datei mit dem Pfadnamen *name* und den Dateiattributen *attr*. Die Dateiattribute sind dabei wie folgt zu kodieren:

Bit 0	*name* kann nur gelesen werden.
Bit 1	*name* ist unsichtbar.
Bit 2	*name* ist eine Systemdatei.
Bit 3	*name* ist ein Disketten-Label.

Nach erfolgreichem Anlegen einer Datei gibt *Fcreate* eine positive Kanalkennung zurück.

Fdelete löscht die Datei name.

Fopen öffnet eine bereits existierende Datei mit dem Pfadnamen *name* zum lesenden und/oder schreibenden Zugriff. Der Wert von *mode* entscheidet dabei über die Zugriffsart. Folgende Werte sind zulässig:

0	nur lesender Zugriff
1	nur schreibender Zugriff
2	Beide Zugriffsarten sind erlaubt.

Ergebnis einer erfolgreichen Ausführung von *Fopen* ist eine positive Kanalkennung, die für nachfolgende Zugriffe benutzt werden muß.

Fdup und ***Fforce*** dienen zur Umlenkung von Ein- und Ausgabe auf andere Geräte als die Massenspeicher. Mit *Fdup* wird so etwa ein Zugriff auf eines der folgenden Geräte ermöglicht:

0	Konsoleingabe (stdin)
1	Konsolausgabe (stdout)
2	serielle Schnittstelle (AUX:)
3	parallele Schnittstelle (PRT:)

Unter Angabe der oben genannten Zahlenwerte für das entsprechende Gerät im *stdhandle* erzeugt *Fdup* eine Kanalkennung (*handle*), die genau wie bei *Fopen* für nachfolgende EA-Operationen genutzt werden kann.

Fforce erlaubt Umlenkungen der mit den Standardkanalkennungen identifizierten Kanäle auf "normale" Datenkanäle. Unter Angabe der Kanalkennung des "normalen" Datenkanals (*handle*) und der Kennung des Standarddatenkanals (*stdhandle*) erfolgt die entsprechende Umlenkung.

Fclose schließt einen zuvor mit der Kanalkennung *handle* geöffneten Kanal.

Fread erlaubt das Lesen von *count*-Bytes aus einer Datei mit der Kanalkennung *handle*. Das Ergebnis des Lesevorgangs wird im Puffer *buf* abgelegt. *rc* enthält dabei die Anzahl der tatsächlich gelesenen Bytes.

Fwrite ist gleichermaßen parametrisiert wie *Fread*. Es erlaubt die zu *Fread* inverse Funktion des Schreibens von Daten.

Fseek ermöglicht beliebige Positionierungen innerhalb einer Datei. Dazu wird *Fseek* mit *offset* angegeben, um wieviele Bytes die Position in der mit *handle* identifizierten Datei geändert werden soll. Mit *mode* wird angegeben, in welcher Art die Position geändert werden soll. *mode* kann folgende Werte annehmen:

0	Änderung relativ zum Dateianfang
1	Änderung relativ zur aktuellen Position
2	Änderung relativ zum Dateiende

Setzen und Abfragen von Dateimerkmalen

ANWENDUNG

error = Fdatime(buf, handle, set);
attr = Fattrib(name, set, attr);
error = Frename(zero, oldname, newname);
error = Fsfirst(name, attr);
error = Fsnext();
addr = Fgetdta();
Fsetdta(ptr);

long addr;
char *ptr, *name, *oldname, *newname;
short attr, zero, handle, set;
unsigned *buf;

BESCHREIBUNG

Die oben genannten Funktionen dienen allesamt zum Lesen oder Ändern eines oder mehrerer der folgenden Dateimerkmale:

- Modifikationszeit
- Dateiattribute
- Dateiname

Mit ***Fdatime*** wird die Modifikationszeit einer Datei bearbeitet. Vor Anwendung von *Fdatime* muß die gewünschte Datei zunächst geöffnet werden (*Fopen*), da *Fdatime* im Gegensatz zu den folgenden Funktionen eine Kanalkennung anstelle eines Pfadnamens zur Identifizierung der Datei benötigt. Unter dieser Voraussetzung wird mit *set* angegeben, ob die Modifikationszeit gesetzt (*set* hat den Wert 1) oder gelesen (*set* hat einen Wert ungleich 1) werden soll. *buf* ist dabei die Adresse eines vier Byte großen Puffers, der im ersten Wort die Modifikationszeit enthält. Im zweiten Wort steht das Modifikationsdatum. Die Kodierung der beiden Werte erfolgt dabei genau wie unter "Setzen und Abfragen der Systemzeit" beschrieben.

Fattrib ermittelt/setzt die Dateiattribute der Datei name. In Abhängigkeit von *set* werden die Dateiattribute entweder auf den Wert von *attr* gesetzt (*set* hat den Wert 1) oder als Funktionsergebnis von *Fattrib* berechnet (*set* hat einen Wert ungleich 1). Folgende Dateiattribute können ermittelt/gesetzt werden:

Bit 0	*name* kann nur gelesen werden.
Bit 1	*name* ist unsichtbar.
Bit 2	*name* ist eine Systemdatei.
Bit 3	*name* ist ein Disketten-Label.
Bit 4	*name* ist ein Verzeichnis.
Bit 5	*name* wurde beschrieben und korrekt geschlossen.

Anmerkung: Beim Setzen von Dateiattributen sind nicht alle Variationen sinnvoll und möglich, beispielsweise ist es nicht möglich, an eine Datei das Attribut "Verzeichnis" zu vergeben.

Das dritte Dateimerkmal, der Dateiname, kann mit ***Frename*** geändert werden. Unter Angabe des alten (*oldname*) und des neuen Dateinamens (*newname*) sowie eines Dummy-Wertes (*zero* muß auf jeden Fall 0 sein) erfolgt die gewünschte Änderung.

Vier weitere Funktionen tragen nun dafür Sorge, daß man alle Dateimerkmale auch geschlossen und für mehr als eine Datei auf einmal abfragen kann:

Fsfirst liest eine Beschreibung der ersten Datei mit dem Pfadnamen name und den Dateiattributen *attr* ein. Die Beschreibung wird im sogenannten DTA-Puffer (**D**isk **T**ransfer **A**ddress) abgelegt. Der Pfadname darf dabei im Bereich des Dateinamens das Zeichen '?' für einen unbekannten Buchstaben und das Zeichen '*' für einen unbekannten Teil-String enthalten. Die Dateiattribute sind wie unter *Fattrib* beschrieben zu spezifizieren. Die ermittelten Informationen umfassen insgesamt 44 Bytes und werden wie folgt im DTA-Puffer abgelegt:

Byte 0-20	reserviert	(Typ char[21])
Byte 21	Dateiattribute	(Typ char)
Byte 22-23	Modifikationszeit	(Typ unsigned)
Byte 24-25	Modifikationsdatum	(Typ unsigned)
Byte 26-29	Dateigröße	(Typ long)
Byte 30-43	Dateiname	(Typ char[14])

Die Kodierung von Modifikationszeit und -datum ist unter "Setzen und Abfragen der Systemzeit" beschrieben.

Mit der Verwendung der beiden Zeichen '?' und '*' im Pfad *name* wird die Suche nach Dateimerkmalen im Normalfall nicht eindeutig, so daß *Fsfirst* nur die erste mögliche Datei findet. In diesem allgemeinen Fall können weitere Dateien, mit Suchkriterien gemäß *Fsfirst*, mit der Funktion *Fsnext* abgerufen werden. Zusätzliche Parameter sind dazu nicht erforderlich, da mit *Fsnext* nur eine bereits begonnene Suche weitergeführt werden kann.

Mit den beiden letztgenannten Funktionen ist es kein Problem, Dateimerkmale in den DTA-Puffer zu laden. Aber wo befindet sich dieser? Die Antwort liefert ein weiterer Betriebssystemaufruf: *Fgetdta*. *Fgetdta* ist parameterlos und liefert die gewünschte Adresse. Da dieser Puffer aber möglicherweise bereits von anderen Programmteilen genutzt wird, ist es sinnvoll, den Betriebssystemaufruf *Fsetdta* zu gebrauchen, um einen eigenen DTA-Puffer zu verwenden. Der einzige Parameter der Funktion *Fsetdta* ist die Adresse des gewünschten Puffers. Das normale Vorgehen zum Umgang mit dem DTA-Puffer sieht daher auch so aus, daß man zunächst die Adresse des alten DTA-Puffers mit *Fgetdta* erfragt und sichert. Danach wird ein eigener DTA-Puffer mit Hilfe von *Fsetdta* bekanntgemacht. Es werden die gewünschten *Fsfirst*- und *Fsnext*-Anfragen durchgeführt. Abschließend wird die alte, zuvor gesicherte Adresse des DTA-Puffers wieder mit *Fsetdta* zum aktuellen Puffer.

Anmerkung: Tatsächlich liegt der DTA-Puffer nach dem Start eines Programms außerhalb des im User-Betrieb erreichbaren Bereichs. Wenn man trotzdem mit dem voreingestellten DTA-Puffer arbeiten will, ist es erforderlich, zuvor in den Supervisor-Betrieb umzuschalten (siehe weiter unten).

Bearbeiten von Verzeichnissen

ANWENDUNG

```
error = Dcreate(path);
error = Ddelete(path);
error = Dsetpath(path);
error = Dgetpath(path, drive);

short error, drive;
char *path;
```

BESCHREIBUNG

Die vier obigen Funktionen dienen zur Verwaltung der Verzeichnisstruktur eines Dateisystems. Im einzelnen besitzen sie folgende Bedeutung:

Dcreate erzeugt ein neues Unterverzeichnis mit dem Pfad *path*.

Ddelete löscht ein vorhandenes, leeres Unterverzeichnis mit dem Pfad *path*.

Dsetpath macht path zum Arbeitsverzeichnis des aktuellen Geräts.

Dgetpath erfragt das Arbeitsverzeichnis des Geräts *drive*. Enthält *drive* dabei den Wert 0, wird das Arbeitsverzeichnis des aktuellen Geräts abgefragt. *drive* = 1 steht für das Gerät A, *drive* = 2 für das Gerät B etc.

Setzen und Abfragen der Systemzeit

ANWENDUNG

```
date = Tgetdate();
Tsetdate(date);
time = Tgettime();
Tsettime(time);

unsigned date, time;
```

BESCHREIBUNG

Tsetdate und ***Tsettime*** setzen das Systemdatum und die -zeit auf die in *date* und *time* kodierten Werte. *Tgetdate* und *Tgetdate* lesen das Systemdatum und die -zeit in date und time ein. *date* ist dabei wie folgt kodiert:

Bit 0-4	Tag
Bit 5-8	Monat
Bit 9-15	Jahr (Offset 1980)

Für time gilt entsprechend:

Bit 0-4	Sekunde (in 2-Sekunden-Intervallen)
Bit 5-10	Minute
Bit 11-15	Stunde

Dynamische Speicherverwaltung

Zum Verständnis der folgenden Ausführungen ist zuvor der Begriff des *Programm-Heaps* zu erläutern: Unter einem Programm-Heap versteht man einen beim Start des Programms ungenutzten Speicherbereich, der zusätzlichen Speicher für eine Vergabe während der Laufzeit des Programms bereithält.

ANWENDUNG

```
addr = Malloc(size);
error = Mfree(addr);
error = Mshrink(zero, mem, size);

long addr, size;
short error, zero;
char *mem;
```

BESCHREIBUNG

Malloc fordert *size* Bytes freien Speicher vom Programm-Heap an. Kann der freie Speicher zur Verfügung gestellt werden, wird die Startadresse des reservierten Bereichs zurückgegeben, ansonsten 0. Eine weitere Funktion von *Malloc* besteht darin, die Gesamtgröße des noch freien Speichers zu erfragen. Dies geschieht, indem *Malloc* mit einem Wert von -1L für *size* aufgerufen wird. In diesem Fall wird in *addr* die gewünschte Größe abgelegt.

Mit ***Mfree*** wird ein durch *Malloc* vergebener Speicherbereich wieder freigegeben.

Mshrink ändert die Größe des Programm-Heaps. *zero* muß dabei immer 0 sein. *mem* muß auf den Speicherbereich zeigen, der für nachladbare Programme reserviert ist. *size* muß die Gesamtgröße des Programms (inklusive statischer Daten und Programm-Heap) angeben. *Mshrink* wird normalerweise nicht direkt vom C-Programm aufgerufen. Sein Aufruf erfolgt automatisch vor Ausführung der ersten Anweisungen in *main*.

Prozeßverwaltung

ANWENDUNG

```
code = Pexec(mode, path, commandline,
             environment);
Pterm0();
Pterm(code);
Ptermres(keep, code);
long code, keep;
short mode;
char *path, *commandline,
            *environment;
```

BESCHREIBUNG

Mit ***Pexec*** kann ein weiteres Programm (*path*) nachgeladen und/oder gestartet werden. Dabei besteht die Möglichkeit, sowohl eine Kommandozeile (*commandline*) als auch Umgebungsvariablen (*environment*) zu übergeben. Sollen die Umgebungsvariablen des aufrufenden Programms genutzt werden, ist *NULL* für environment zu übergeben. Mit *mode* wird angegeben, welche Operation mit *Pexec* durchgeführt werden soll. Folgende Werte sind dabei möglich:

0	Laden und Starten
3	nur Laden
4	nur Starten
5	nur eine Basisseite erzeugen und freien Speicher zuweisen

Bei *mode = 0* ist *path* der Pfadname des zu ladenden Programms. Das Ergebnis von Pexec ist der *exit*-Status des ausgeführten Programms. Bei *mode = 3* ist *path* ebenfalls der Pfadname des Programms. Das Ergebnis von *Pexec* ist jedoch die Adresse der Basisseite des geladenen (und nicht ausgeführten) Programms. Bei *mode = 4* ist *path* die Adresse der Basisseite.

Ferner ist zu beachten, daß *commandline* eine Zeichenkette im Pascal-Format ist, d.h. im ersten Byte wird die Länge der Zeichenkette angegeben, die Daten für die Zeichen folgen und müssen nicht nullterminiert sein.

Die drei Funktionen ***Pterm0***, ***Pterm*** und ***Ptermres*** sind Varianten des Programmabbruchs. *Pterm0* beendet ein Programm mit dem *exit*-Status 0. *Pterm* und *Ptermres* dagegen beenden ein Programm mit *exit*-Status *code*. Bei *Ptermres* besteht zusätzlich zur Angabe eines *exit*-Status noch die Möglichkeit, einen Teil oder das gesamte Programm nach der Programmausführung im Hauptspeicher zu behalten. Dazu ist mit *keep* die Größe des Speicherbereichs anzugeben, der nach der Programmausführung im Hauptspeicher gehalten werden soll. Damit ist es z.B. möglich, ein Programm mehrfach zu starten, ohne es mehrfach zu laden. Zusätzlich gilt, daß alle drei Funktionen einen kontrollierten Programmabbruch durchführen, d.h. eventuell offene Kanäle werden automatisch vor dem endgültigen Exitus geschlossen.

Anmerkung: Von mir wird zum kontrollierten Programmabbruch keine dieser drei Funktionen verwendet. Stattdessen werden Sie an gegebener Stelle die UNIX-kompatible Funktion *exit* vorfinden. Im Prinzip macht sie allerdings das gleiche wie *Pterm* und ihre Varianten.

Diverses

Für zwei Funktionen konnte ich keine sinnvolle Zuordnung zu einer funktionalen Gruppe finden. Sie sind hier als "Diverses" zusammengefaßt.

ANWENDUNG

```
oldstack = Super(newstack);
version  = Sversion();

long oldstack, newstack;
short version;
```

BESCHREIBUNG

Der 68000-Prozessor des ST kennt zwei Betriebsmodi: den *User*- und den *Supervisor-Modus*. Im User-Modus, der während eines "normalen" Programmablaufs aktiv ist, sind keine direkten Zugriffe außerhalb des Adreßraumes eines Programms erlaubt. Direkte Zugriffe auf Adressen außerhalb dieses Adreßraumes, hier liegen beispielsweise die Systemvariablen u.a., verursachen einen Fehler, der zum Programmabsturz führt. Im Supervisor-Modus existiert dieser Zugriffsschutz nicht. Die Funktion *Super* kann nun dazu genutzt werden zwischen den beiden Betriebsmodi hin- und herzuschalten. Beim ersten Aufruf von *Super* (normalerweise im User-Modus) ist dazu ein Zeiger auf einen freien Speicherbereich zu übergeben (*newstack*), der im Supervisor-Modus als Stack benutzt werden soll. Als Ergebnis erhält man einen Zeiger auf den User-Stack zurück, den es für den Fall aufzubewahren gilt, daß wieder mit *Super* in den User-Modus zurückgeschaltet werden soll. Eine weitere Funktion von *Super* ist die Abfrage des momentan vorliegenden Modus'. Erhält *Super* den Wert -1L als Parameter, gibt sein Ergebnis darüber Aufschluß, in welchem Modus sich der 68000 befindet. Ein Wert von 0 bedeutet in diesem Fall "User-Modus", ein Wert von 1 dagegen "Supervisor-Modus".

Zur Abfrage der GEMDOS-Versionsnummer existiert die Funktion *Sversion*. Sie liefert das Wort *version*, das im niederwertigen Byte die Versionsnummer und im höherwertigen die Nummer der Überarbeitung enthält.

Vorausschau

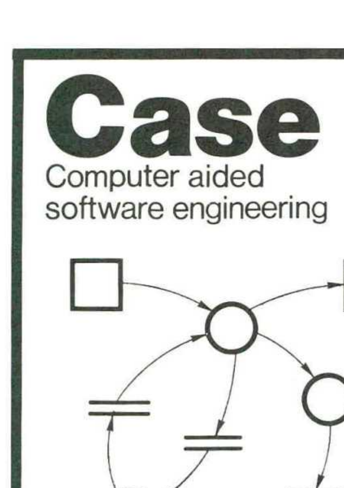
So, das war gewissermaßen die Pflicht. Das GEMDOS ist nun dokumentiert, und in der nächsten Folge der "Programmer's Toolbox" kann mit der Kür, der Programmierung, begonnen werden. Zunächst wird dann das Modul ATOM programmiert. Es übernimmt so elementare Aufgaben wie das Kopieren, das Bewegen und das Löschen von Dateien und bildet daher eine weitere Grundlage bei der Programmierung der Kommandos.

Dirk Brockhaus



Case

Computer aided software engineering



Structured Analysis

- Dataflow Diagram Editor
- Specification Editor
- Data Dictionary

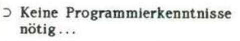
integrierte Funktionen
konsistente Datenbasis

für:	
ATARI-ST	290 DM
IBM-AT	390 DM
DemoVersion	25 DM

SOFTWARETECHNIK
Dipl.-Ing. U. Böhnke
Lindenstr. 31
8012 Ottobrunn
Tel. 089 / 609 2449

Ein Traum wird wahr – mit HASCS - THE GAME CREATOR erhalten Sie die Möglichkeit, Ihre eigene Rollen- und Fantasyspiele zu entwickeln. Sie müssen keine Programmiersprache beherrschen – Ihre Fantasie reicht vollkommen aus!





- Keine Programmierkenntnisse nötig...
- Monster, Landschaften, Charaktere, Waffen etc. nach eigenen Wünschen erstellbar...
- Spezieller Sprite-Editor (HASCS - Sprites) liegt bei...
- Dialoge (zwischen Spieler und Charakteren, Monstern) mit unterschiedlichsten Sprachschätzen möglich...
- Ein Experimentierspiel (HASCS - Rosa von Rosendorn) liegt bei...
- Entwickelte Spiele dürfen getauscht, verkauft oder auf andere Art vertrieben werden...

HASCS - THE GAME CREATOR läuft nur in hoher Auflösung; zweiseitiges Laufwerk erforderlich. Ein ausführliches, verständlich geschriebenes Handbuch liegt bei!



Händleranfragen erwünscht.

• HASCS ist ein Rollenspielentwicklungs-System. Es gibt bereits sehr bekannte Spiele, die mit diesem System entwickelt wurden (z.B.: "Rat der Weisen"; siehe hierzu Bericht in der Zeitschrift "PD-JOURNAL" Ausgabe 4/90, Seite 33).

HASCS - THE GAME CREATOR ... HASCS - Editor, HASCS - Sprites, HASCS - Run Only, HASCS - Spiel

nur DM 59,-- (zuzügl. DM 3,-- Porto)

Bezahlung: per Nachnahme, Barzahlung oder Scheck

Das komplette HASCS - System erhalten Sie von...

COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33 B, 7000 Stuttgart 80
Telefon: 07 11 / 6 78 73 92

Sollten Sie Fragen zu dem neuartigen HASCS - System haben, so senden wir Ihnen kostenloses Informationsmaterial zu.

iks

SM 124 Multisync II

Bausatz zum Umrüsten von Ihrem Monochrommonitor zur Darstellung der 3 Auflösungen.

Platine incl. Kabel und allen Bauteilen	**DM 129,00**
Platine fertig bestückt	**DM 149,00**
Umbau Ihres Monitors	**DM 249,00**

7 Tage Rückgaberecht für die Bausätze. Preise incl. Testbild und Malprogramm auf Diskette.
Bei Bestellung bitte Baujahr angeben.

Public Domain

Serie 2000 PD Pool	**DM 8,00**
Serie "ST Comp." ab 6 Stück	**DM 5,00**
bis 5 Stück	**DM 6,00**

Wir liefern bis PD 223/224 immer 2 PD's auf einer Disk. (ungerade beginnend und aufeinanderfolgend)
Ausführliche Liste und Zeitschrift PD-Szene gegen Rückporto DM 1,50.

Preise zzgl. Porto und Verpackung DM 5,00, NN plus DM 2,00, besser V-Scheck (Ausland nur Euro-Scheck).

iks
In der Au 22
7516 Karlsbad 4
07202/7687



Dieter Geiß · Jürgen Geiß
Vom Anfänger zum GEM-Profi
Perfekte Programmierung auf Atari ST und IBM-PC
Hüthig

Dieter und Jürgen Geiß
Vom Anfänger zum GEM-Profi
Perfekte Programmierung auf Atari ST und IBM-PC
1990, 532 S., geb., DM 98,– incl. Diskette
ISBN 3-7785-1792-9

Eine komplette Programmierumgebung für GEM-Software, sodaß Entwickler nur noch minimalen Aufwand betreiben müssen, um perfekte Software unter GEM zu erstellen. Ermöglicht z.B. Pop-Up-Menüs oder Menüzeilen in Fenstern, Accessories mit eigenem Desktop und Menüzeile. Die komplexe Steuerung unter einer graphischen Oberfläche wird dem Programmierer komplett abgenommen. Mit Richtlinien zur GEM-Programmierung, empfohlen von Atari und Digital Research.



Christiane und Jürgen Kehrel
Omikron-BASIC
Befehle, Bibliotheken, Utilities
1989, 590 S., geb., DM 58,–
ISBN 3-7785-1662-0

Nach der Handhabung des Editors wird jeder BASIC-Befehl (auch die undokumentierten) präzise vorgestellt und sowohl mit einem Syntaxdiagramm als auch einem Beispiel näher erklärt. Es folgen BIOS-, XBIOS- und GEMDOS-Aufrufe mit einer Funktions- und Parameterbeschreibung. Auch die komplette GEM.LIB (AES und VDI) wird auf 150 Seiten so beschrieben, wie sie wirklich unter OMIKRON.BASIC ansprechbar ist. Erläuterungen der wichtigsten Bibliotheken (z.B. EasyGEM) und der verfügbaren Hilfsprogramme runden das Buch ab, in dem dank der mehr als 3000 Verweise im Register jeder Begriff schnell gefunden werden kann.

BESTELLCOUPON

einsenden an: Hüthig Buch Verlag GmbH
Im Weiher 10, 6900 Heidelberg

Titel

Name, Vorname

Straße, Nr.

PLZ, Ort

Datum, Unterschrift



FPU-Precompiler

für Modula-2

Teil 2

Nachdem in der letzten Folge die Fähigkeiten des Coprozessors 68881 behandelt wurden, soll es diesmal darum gehen, wie man mit einem Precompiler den Coprozessor aus Modulaprogrammen heraus aufrufen kann. Daneben soll auch der Precompiler beschrieben werden, denn wer Milch trinkt, sollte auch wissen, wie eine Kuh aussieht (in Abwandlung eines Spruchs von Steven Jobs). Schließlich soll dann aus der Milch Fruchtjoghurt werden: ein optimiertes Programm.

Wie sag' ich's ihm?

An dieser Stelle soll nun erläutert werden, wie man die Arithmetikoperationen, die in der letzten Ausgabe vorgestellt wurden, notieren muß, damit sie der Precompiler in Maschinencode umsetzen kann. Zunächst muß der Precompiler wissen, mit welchem Compiler er eingesetzt wird, damit er sich bei seiner Arbeit darauf einstellen kann. Deshalb muß irgendwo, z.B. in einem Kommentar, entweder "Hänisch", "TDI" oder "TU München" stehen. Da man ohnehin irgendwo angeben sollte, mit welchem Compiler man arbeitet, ist dies wohl die einfachste Lösung. Das grundsätzliche Format der Operationen ist einheitlich. Eine typische Zeile könnte z.B. wie folgt aussehen:

```
FsinD (argument, fp0) ; (* Berechnung der Winkelfunktion *)
```

Die Zeile besteht aus folgenden Komponenten:

F:
Dieses 'F' soll anzeigen, daß eine Anweisung für den Precompiler vorliegt. Dies trägt zum einen dazu bei, daß man die Bezeichner für die Operationen auch andersweitig verwenden kann, z.B. für komplexe Funktionen aus einer Bibliothek, und vereinfacht zum anderen dem Precompiler die Suche. Das 'F' muß immer groß geschrieben werden.

sin:
Dies ist die eigentliche Operation, die immer klein geschrieben wird.

D:
Das 'D' gibt die Genauigkeit an, mit der die Variable aus dem Speicher vorliegt. Der Precompiler unterstützt die Variablenbreiten B bis E, nicht jedoch P, da dieses BCD-Format extrem zeitaufwendig und somit für Hochgeschwindigkeitsrechnungen ohnehin uninteressant ist. Diese Angabe wird wiederum groß geschrieben und sollte direkt an die Operation anschließen. Macht man keine Angaben zur Genauigkeit, geht der Precompiler davon aus, daß er es mit einer LONGREAL-Variable zu tun hat, was wohl der Regelfall sein dürfte. Diese Konvention erspart einem auf Dauer doch einige Tipperei. Bei den Register-Register-Operationen erübrigt sich natürlich eine Angabe der Rechengenauigkeit, da hier ohnehin immer mit den internen 80 Bits gerechnet wird.

(argument, fp0):
Die nächste Angabe ist die Argumentenliste. Hier kann beliebig Freiraum eingefügt werden. Die Argumente werden durch Kommata getrennt. Die Argumentenliste enthält je nach Operation zwischen null (dann wird sie ignoriert) und vier Argumente. Das Gros der Anweisungen hat zwei Argumente. Das erste Argument ist der Quell,-das zweite der Zieloperand. Die davon abweichenden Operationen werden im nächsten Abschnitt behandelt. Der Quelloperand kann auch eine Zahl sein, die dann gemäß der Genauigkeitsangabe umgewandelt und aus dem Speicher in die FPU übertragen wird. Dabei wird die Genauigkeit E allerdings nicht unterstützt, da man so genaue Zahlen ohnehin nicht hat und es auch keine allgemein zugänglichen Umwandlungsroutinen von Strings nach E gibt. Variablen können selbstverständlich auch mit der maximalen Genauigkeit übertragen werden, was allerdings wohl nur bei Zwischenwerten von langen Rechnungen sinnvoll sein dürfte. Konstanten, die irgendwo, eventuell in einem anderen Modul

mit CONST definiert wurden, lassen sich nicht auswerten. Sie würden als Variablennamen interpretiert werden, und der Compiler wäre dann nicht in der Lage, dafür eine Variablenadresse anzugeben. Eine Ausnahme bilden lediglich Konstanten des Typs LONGINT, die als 32 Bit lange Zahlen direkt geladen werden.

Ebenfalls nicht als Quelloperand geeignet sind Funktionsaufrufe. Der Precompiler müßte dann erst wieder den Weg über eine Variable gehen, da er ja nicht weiß, wo die Funktion im Speicher stehen wird, und das kann der Benutzer ja selbst erledigen.

;:
Das Semikolon ist nicht zwingend, aber da es bei den Hochsprachen halt so üblich ist, wurde dafür gesorgt, daß es auch nicht stört. Man kann auch ohne Semikolon mehrere Precompiler-Anweisungen aneinanderfügen.

(* Berechnung der Winkelfunktion *): Kommentierungen sind natürlich auch möglich. Beim Übersetzen werden die Kommentare jeweils unter die letzte Zeile des äquivalenten Maschinencodes gesetzt, quasi als Abschlußkommentar. Hat man mehrere Precompiler-Anweisungen in einer Zeile, so gelangt ein die Zeile abschließender Kommentar unter die letzte Maschinencodezeile der letzten Operation. An den Kopf der einzelnen Operationen wird jeweils als Kommentar die Anweisung an den Precompiler gesetzt.

Spezielle Anweisungen

Wie der Lateiner weiß, gibt es keine Regel ohne Ausnahme. Ebenso geht es der Regel, daß eine Arithmetikoperation zwei Operanden hat:

RESET, STATUS:

Diese beiden Pseudooperationen haben keine Argumentenliste, da sie keine brauchen. Die Anweisungen werden *Freset;* bzw. *Fstatus;* geschrieben.

Freset bewirkt einen Reset des Coprozessors, wie er z.B. nach einer Division durch Null nötig werden kann, damit der Hauptprozessor wieder mit dem Coprozessor ins Gespräch kommt. Die Fließkommaregister bleiben dabei erhalten. Am Anfang eines jeden Programms sollte ein solcher Reset durchgeführt werden, um klare Verhältnisse herzustellen.

Fstatus liest das Response-Register in eine vom Programmierer als LONGINT zu deklarierende Variable *'fpustatus'*. Die oberen 16 Bits sollten $802 ergeben. Ergeben sie $1D0D, kann man gleich einen Reset folgen lassen, da dann irgendwann eine Protokollverletzung durch Überläufe, einen langsameren Coprozessor oder, was ich nicht hoffe, durch einen Timing-Fehler des Precompilers erzeugt wurde, die sich bis zum nächsten Reset durchzieht. Theoretisch könnte man den Code auch so produzieren, daß bei Protokollverletzungen eine bestimmte Routine aufgerufen wird, die den Benutzer warnt. Solche Abfragen sind allerdings nicht dazu angetan, die Geschwindigkeit zu steigern. Daher wurde diese Pseudooperation eingeführt, damit man an kritischen Stellen selbst eine Prüfung durchführen kann.

SINCOS:

Da hiermit zwei Funktionswerte berechnet werden, gibt es auch zwei Zieloperanden. Motorola legt die Syntax für Assembler so fest, daß erst das Ziel für den Cosinus angegeben wird und dann

```
675:
676:  MODULE Codierung ;
677:    IMPORT fpuop, tfpuop, Operationstyp,
               Extensiontyp, Uebergabetyp,
678:           Vergleichstyp, Compileridentifizieren,
               Compiler, Compilertyp,
679:           WriteString, WriteLn,
680:           Klammerauf, Klammerzu,
681:           InOut, ADR, ADDRESS,
682:           FehlerMeldung, StrToVal, ValToStr,
               StrToReal,
683:           Pos, Length, Copy, Concat ;
684:    EXPORT Codegenerierung ;
685:
686:    PROCEDURE Codegenerierung() : BOOLEAN ;
687:    VAR AnfangZeile,
688:        FlOp,
689:        EndeZeile   : ARRAY [0..511] OF CHAR ;
690:        Einrueckung, i : SHORTCARD ;
691:        Einrueckstring : ARRAY [0..80] OF CHAR ;
            (* sollte wohl reichen *)
692:        CodeWerte : ARRAY [1..6] OF SHORTCARD ;
693:
694:      PROCEDURE TransferZeile ;
695:      VAR Laenge : SHORTCARD ;
696:      BEGIN
697:        Copy(AnfangZeile, fpuop.Zeile, 0,
                 fpuop.ZeigerAnfang) ;
698:        Einrueckung := fpuop.ZeigerAnfang ;
699:        Concat(AnfangZeile, Klammerauf) ;
700:        Copy(FlOp, fpuop.Zeile, fpuop.ZeigerAnfang,
                 fpuop.Zeiger - fpuop.ZeigerAnfang) ;
701:        Concat(AnfangZeile, FlOp) ;
702:        Concat(AnfangZeile, Klammerzu) ;
703:        WriteString(AnfangZeile) ; WriteLn ;
704:        Laenge := Length(fpuop.Zeile) ;
705:        WHILE fpuop.Zeile[fpuop.Zeiger] = " " DO
706:          INC(fpuop.Zeiger) ;
707:        END ;
708:        Copy(EndeZeile, fpuop.Zeile, fpuop.Zeiger,
                 Laenge - fpuop.Zeiger) ;
709:        fpuop.Zeile := "" ;
710:        FOR i := 1 TO Einrueckung DO
711:          Concat(fpuop.Zeile, " ") ;
712:        END ;
713:        Copy(Einrueckstring, fpuop.Zeile, 0,
                 Einrueckung) ;
714:      END TransferZeile ;
715:
716:      PROCEDURE Make1Code(w1 : SHORTCARD) ;
717:      BEGIN
718:        CodeWerte[1] := w1 ;
719:        MakeCode(1) ;
720:      END Make1Code;
721:      PROCEDURE Make2Code(w1, w2 : SHORTCARD) ;
722:      BEGIN
723:        CodeWerte[1] := w1 ; CodeWerte[2] := w2 ;
724:        MakeCode(2) ;
725:      END Make2Code;
726:      PROCEDURE Make3Code(w1, w2, w3 : SHORTCARD) ;
727:      BEGIN
728:        CodeWerte[1] := w1 ; CodeWerte[2] := w2 ;
            CodeWerte[3] := w3 ;
729:        MakeCode(3) ;
730:      END Make3Code;
731:      PROCEDURE Make4Code(w1, w2, w3, w4 :
                              SHORTCARD) ;
732:      BEGIN
733:        CodeWerte[1] := w1 ; CodeWerte[2] := w2 ;
            CodeWerte[3] := w3 ;
734:        CodeWerte[4] := w4 ;
735:        MakeCode(4) ;
736:      END Make4Code;
737:      PROCEDURE Make5Code(w1, w2, w3, w4, w5 :
                              SHORTCARD) ;
738:      BEGIN
739:        CodeWerte[1] := w1 ; CodeWerte[2] := w2 ;
            CodeWerte[3] := w3 ;
740:        CodeWerte[4] := w4 ; CodeWerte[5] := w5 ;
741:        MakeCode(5) ;
742:      END Make5Code;
743:      PROCEDURE Make6Code(w1, w2, w3, w4, w5, w6 :
                              SHORTCARD) ;
744:      BEGIN
745:        CodeWerte[1] := w1 ; CodeWerte[2] := w2 ;
            CodeWerte[3] := w3 ;
746:        CodeWerte[4] := w4 ; CodeWerte[5] := w5 ;
```
→

das Ziel für den Sinus. Da ich mir aber die Reihenfolge eh nicht merken kann, habe ich es umgedreht, um somit die Reihenfolge aus dem Namen der Operation ablesen zu können. Damit alle Programme, die den Precompiler benutzen, übertragbar bleiben, bitte ich darum, an dieser Reihenfolge nichts zu ändern!

CMP, TST: Die Argumentenliste von *Ftst* umfaßt ebenfalls drei Einträge:

- Quelloperand: die zu testende Zahl

- Boolsche Variable: eine Variable, in die geschrieben wird, ob die Bedingung erfüllt ist oder nicht

- Testbedingung: Eine der 32 Möglichkeiten aus Tabelle 3 der letzten Folge.

  Bei *Fcmp* kommt nach dem Quell- noch der Zieloperand hinzu, von dem der Quelloperand zum Vergleich abgezogen wird. Es sei hier nochmals darauf hingewiesen, daß der Zieloperand dabei erhalten bleibt.

Im Gegensatz zur lateinischen Grammatik ist die Liste der Ausnahmen hier bereits zu Ende.

Benötige Importe

In dem produzierten Precompilat, das gemäß Ihren Anweisungen erzeugt wird, befinden sich einige Ausdrücke, die aus dem Modul *SYSTEM* stammen. Sie müssen daher je nach Compiler *ADR, CODE, LOAD* und *STORE* (Hänisch-Modula) bzw. *ADR, INLINE, SETREG* und *REG* (PD-Modula der TU München) oder *ADR, CODE, SETREG* und *REGISTER* (TDI Modula) aus dem Pseudomodul *SYSTEM* importieren. Sollten Sie dies vergessen, so wird Sie Ihr Compiler darauf aufmerksam machen und Sie können es nachträglich verbessern.

Arbeitsweise des Programms

Nachdem die Syntax des Precompilers, die wohl nicht allzu kompliziert ist, behandelt wurde, soll an dieser Stelle seine Arbeitsweise erläutert werden, um einem breiteren Anwenderkreis eine Anpassung an andere Compiler, eine alte 12 MHz-FPU oder eine 16 MHz-CPU zu ermöglichen.
Um das Programm leichter durchschaubar zu machen, wurden die Analyse der Textquelle und die Synthese des Codes getrennt und in den zwei Modulen *Parser* und *Codierung* untergebracht. Ein drittes Modul *StreamHandling* beschäftigt sich mit dem Öffnen und Schließen der benötigten Dateien. Die Module *StreamHandling* und *Parser* wurden bereits in der letzten Ausgabe abgedruckt, das Modul *Codierung* findet sich in diesem Artikel. Auf der Diskette zum Precompiler befindet sich u.a. nochmals der komplette Source-Text in Modula und als Pseudocode, aus dem das Modulaprogramm entwickelt wurde.

Datenstruktur des Precompilers

Die gesamten Informationen, die aus der Eingabezeile kommen, werden in einem RECORD *fpuop* vom Modul *Parser* gesammelt und vom Modul *Codierung* ausgewertet.
Die einzelnen Datentypen dieses Records entnehme man dem Programm-Listing:

```
            CodeWerte[6] := w6 ;
747:        MakeCode(6) ;
748:      END Make6Code;
749:
750:      PROCEDURE MakeCode(n : SHORTCARD) ;
751:      VAR HexCode : ARRAY [0..4] OF CHAR ;
752:          IntWert : INTEGER ;
753:          i       : SHORTCARD ;
754:          error   : BOOLEAN ;
755:      BEGIN
756:        Copy(fpuop.Zeile, Einrueckstring, 0,
                 Einrueckung) ;
757:        CASE Compiler OF
758:          HaenischM, Megamax :
759:            FOR i := 1 TO n DO
760:              Concat(fpuop.Zeile, "CODE(0") ;
761:              IntWert := CodeWerte[i] ;
762:              ValToStr(IntWert, FALSE, HexCode, 16,
                           4, "0", error) ;
763:              Concat(fpuop.Zeile, HexCode) ;
764:              Concat(fpuop.Zeile, "H);") ;
765:            END |
766:          PDM, TDI, SPC :
767:            IF Compiler = TDI
768:              THEN Concat(fpuop.Zeile, "CODE(") ;
769:              ELSE Concat(fpuop.Zeile, "INLINE(") ;
770:            END ;
771:            FOR i := 1 TO n DO
772:              IF i # 1
773:                THEN Concat(fpuop.Zeile, ",") ;
774:              END ;
775:              IntWert := CodeWerte[i] ;
776:              ValToStr(IntWert, FALSE, HexCode, 16,
                           4, "0", error) ;
777:              Concat(fpuop.Zeile, "0") ;
778:              Concat(fpuop.Zeile, HexCode) ;
779:              Concat(fpuop.Zeile, "H") ;
780:            END ;
781:            Concat(fpuop.Zeile, ");") ;
782:        END ;
783:        WriteString(fpuop.Zeile) ; WriteLn ;
            INC(fpuop.Zeilennummer) ;
784:      END MakeCode ;
785:
786:      PROCEDURE LoadAdr(var : ARRAY OF CHAR) ;
787:      BEGIN
788:        Copy(fpuop.Zeile, Einrueckstring, 0,
                 Einrueckung) ;
789:        CASE Compiler OF
790:          HaenischM, Megamax :
791:            Concat(fpuop.Zeile, "LOAD(ADR(") ;
792:            Concat(fpuop.Zeile, var) ;
793:            Concat(fpuop.Zeile, "),8);") |
                  (* 8 für a0 *)
794:          PDM, TDI, SPC :
795:            Concat(fpuop.Zeile, "SETREG(8,ADR(") ;
                  (* 8 für a0 *)
796:            Concat(fpuop.Zeile, var) ;
797:            Concat(fpuop.Zeile, "));") |
798:        END ;
799:        WriteString(fpuop.Zeile) ; WriteLn ;
            INC(fpuop.Zeilennummer) ;
800:      END LoadAdr ;
801:
802:      PROCEDURE LoadData ;
803:      BEGIN
804:        Copy(fpuop.Zeile, Einrueckstring, 0,
                 Einrueckung) ;
805:        CASE Compiler OF
806:          HaenischM, Megamax :
807:            Concat(fpuop.Zeile, "LOAD(") ;
808:            Concat(fpuop.Zeile, fpuop.VarNum) ;
809:            Concat(fpuop.Zeile, ", 0);") |
                  (* 0 für d0 *)
810:          PDM, TDI, SPC :
811:            Concat(fpuop.Zeile, "SETREG(0,") ;
                  (* 0 für d0 *)
812:            Concat(fpuop.Zeile, fpuop.VarNum) ;
813:            Concat(fpuop.Zeile, ");") ;
814:        END ;
815:        WriteString(fpuop.Zeile) ; WriteLn ;
            INC(fpuop.Zeilennummer) ;
816:      END LoadData ;
817:
818:  (*    PROCEDURE StoreData ;
819:      BEGIN
820:        Copy(fpuop.Zeile, Einrueckstring, 0,   →
```

```
          Einrueckung) ;
821:      CASE Compiler OF
822:        HaenischM, Megamax :
823:          Concat(fpuop.Zeile, "STORE(0,") ;
824:          Concat(fpuop.Zeile, fpuop.VarNum) ;
825:          Concat(fpuop.Zeile, ");") |
826:        PDM, SPC :
827:          Concat(fpuop.Zeile, fpuop.VarNum) ;
828:          Concat(fpuop.Zeile, ":=REG(0);") |
829:        TDI :
830:          Concat(fpuop.Zeile, fpuop.VarNum) ;
831:          Concat(fpuop.Zeile, ":=REGISTER(0);") ;
832:      END ;
833:      WriteString(fpuop.Zeile) ; WriteLn ;
          INC(fpuop.Zeilennummer) ;
834:    END StoreData ;
835:  *)
836:    PROCEDURE Nops(AnzNops : SHORTCARD) ;
837:    BEGIN
838:      IF AnzNops <= 6
839:        THEN FOR i := 1 TO AnzNops DO
840:               CodeWerte[i] := 04E71H ;
841:             END ;
842:             IF AnzNops > 0
843:               THEN MakeCode(AnzNops) ;
844:             END ;
845:        ELSE Make4Code(041F8H, 0FA41H, 04A10H,
                          06FCH) ;
846:             (* lea $FA41.w, a0 ; tst.b (a0) ;
                 beq.s zu tst.b (a0) *)
847:      END ;
848:    END Nops ;
849:
850:    PROCEDURE ResponseTest ;
851:    BEGIN
852:      Make2Code(04A38H, 0FA41H) ;
          (* tst.b $fa41.w *)
853:    END ResponseTest ;
854:
855:    PROCEDURE RechenCode() : SHORTCARD ;
856:    VAR Code : SHORTCARD ;
857:        romoffsetl  : INTEGER ;
858:        romoffsets  : SHORTCARD ;
859:        sign, error :BOOLEAN ;
860:    BEGIN
861:      IF fpuop.Argument2 < Reg0
862:        THEN Code := 3 * 8192 ;
863:      ELSIF fpuop.Argument1 < Reg0
864:        THEN Code := 2 * 8192 ;
865:        ELSE Code := 0 * 8192 ;
866:      END ;
867:      IF NOT( (fpuop.Operation = movecr) OR
                  (fpuop.Operation = sincos) )
868:        THEN CASE fpuop.Operation OF
869:               move    : Code := Code + 0 |
870:               int     : Code := Code + 1 |
871:               sinh    : Code := Code + 2 |
872:               intrz   : Code := Code + 3 |
873:               sqrt    : Code := Code + 4 |
874:               lognp1  : Code := Code + 6 |
875:               etoxm1  : Code := Code + 8 |
876:               tanh    : Code := Code + 9 |
877:               atan    : Code := Code + 10 |
878:               asin    : Code := Code + 12 |
879:               atanh   : Code := Code + 13 |
880:               sin     : Code := Code + 14 |
881:               tan     : Code := Code + 15 |
882:               etox    : Code := Code + 16 |
883:               twotox  : Code := Code + 17 |
884:               tentox  : Code := Code + 18 |
885:               logn    : Code := Code + 20 |
886:               log10   : Code := Code + 21 |
887:               log2    : Code := Code + 22 |
888:               abs     : Code := Code + 24 |
889:               cosh    : Code := Code + 25 |
890:               neg     : Code := Code + 26 |
891:               acos    : Code := Code + 28 |
892:               cos     : Code := Code + 29 |
893:               getexp  : Code := Code + 30 |
894:               getman  : Code := Code + 31 |
895:               div     : Code := Code + 32 |
896:               mod     : Code := Code + 33 |
897:               add     : Code := Code + 34 |
898:               mul     : Code := Code + 35 |
899:               sgldiv  : Code := Code + 36 |
900:               rem     : Code := Code + 37 |
901:               scale   : Code := Code + 38 |    →
```

```
tfpuop = RECORD
       Operation      : Operationstyp ;
       Extension      : Extensiontyp ;
       Argument1      : Uebergabetyp ;
       Argument2      : Uebergabetyp ;
       Argument3      : Uebergabetyp ;
       Argument4      : Vergleichstyp ;
       VarNum         : ARRAY [0..79] OF CHAR ;
       Boolean        : ARRAY [0..79] OF CHAR ;
       Zeile          : ARRAY [0..511] OF CHAR ;
       Zeilennummer   : INTEGER ;
       Zeiger         : SHORTCARD ;
       ZeigerAnfang   : SHORTCARD ;
END ;
```

Tabelle 1: zentrales Daten-Record des Precompilers

Hier ist nur erwähnenswert, daß jeder der Aufzählungstypen auch ein Element enthält, das für einen leeren Eintrag steht. In *VarNum* wird der Bezeichner für eine Variable gespeichert, *Boolean* enthält den Namen der Boolschen Variablen, die beim Vergleich benutzt wird, Zeile ist eine Pufferung für Ein- und Ausgaben, aus der heraus analysiert und in die hinein synthetisiert wird. *Zeiger* wird dabei als aktueller Zeiger benutzt, und in *ZeigerAnfang* merkt sich das Programm, wo die Operation begann. Zeilennummer schließlich wird für Fehlermeldungen benötigt. Die anderen Feldnamen dürften selbstredend gewählt sein.

Analyse der Precompiler-Anweisung

Bevor man etwas analysieren kann, muß man es erst einmal haben. Das Programm sucht daher zunächst nach einem 'F', das außerhalb einer Kommentarklammerung steht. Die Klammern sollen dabei beliebig tief geschachtelt sein können, denn man will ja dem Programmierer keine Einschränkungen auferlegen, sondern neue Möglichkeiten eröffnen. Hierzu dient der Prozedur *Fsuchen()* die Prozedur *Klammernumgehen()*, die erst dann wieder zurückkehrt, wenn sie das Ende der Klammerschachtelung gefunden hat, bzw. wenn das File-Ende erreicht wurde. *Fsuchen()* gibt dem Hauptprogramm zurück, ob es etwas gefunden hat oder nicht. Hierbei liest es so lange neue Zeilen ein, bis es etwas gefunden hat oder das File-Ende erreicht ist.

Als nächstes läßt das Hauptprogramm dann die Prozedur *Operationidentifizieren()* auf den Text los, die mit der Hilfsfunktion *IdentifiziereString()* die Liste der definierten Operationen abklappert. Bei Erfolg wird ein Eintrag in *fpuop. Operation* gemacht, ansonsten wird FALSE an das Hauptprogramm gemeldet, damit die Suche weitergehen kann.

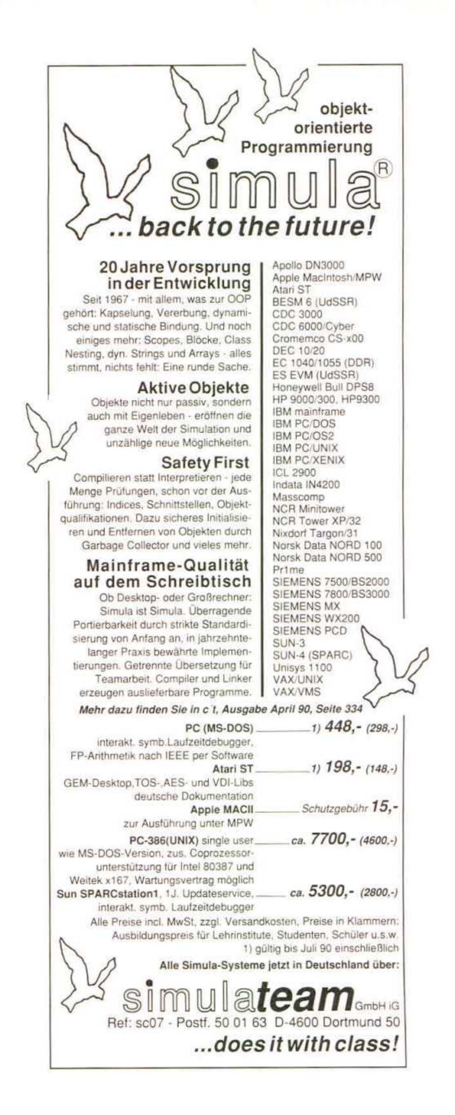
Hat die Funktion keine Argumentenliste, wie die Anweisungen *reset* und *status*, erfolgt hier bereits der Abbruch derAnalyse.

Ansonsten wird mit *Extensionidentifizieren()* zunächst die Genauigkeit einer eventuellen Speichervariablen oder Zahl bestimmt. Liegt etwas nicht Definiertes vor, oder wurde die Datenform P gewählt, gibt es eine entsprechende Fehlermeldung.

Als nächstes wird dann mit *SucheOperand1()* nach der Argumentenliste Ausschau gehalten und untersucht, ob sie auch korrekt ist, d.h. ob die Klammer auch wieder geschlossen wird und etwas in ihr enthalten ist. Bei Erfolg ist der *fpuop.Zeiger* hernach auf den ersten Operanden ausgerichtet. Jetzt beginnt es für *OperandenIdentifizieren()* interessant zu werden. Diese Prozedur sucht nach den Operanden und identifiziert sie. Dabei werden auch eine ganze Reihe an Fehlern moniert, die sofort erkennbar sind. Die Prozedur hat die lokale Prozedur *TestRegVar()*, die einen String als Register bzw. als Variable identifizieren kann, die Prozedur *TestVgl()*, die überprüft ob eine gültige Bedingung vorliegt, und wenn ja, welche, und schließlich die









Wilhelm Mikroelektronik zum Thema Bildverarbeitung und Vektorisierung

Graustufenscanner: (alle Scanner inkl. unten aufgeführter Software)

Charly: echte 400 dpi Handscanner, 105 mm breit, 32 Graustufen	DM 598,–
Charly Page: 400 dpi-DIN A4-Scanner, 64 Graustufen, ideal auch für Buchvorlagen	DM 1498,–
FS-4SF: Vollautomatischer Einzelblatteinzug für den Charly Page	DM 798,–

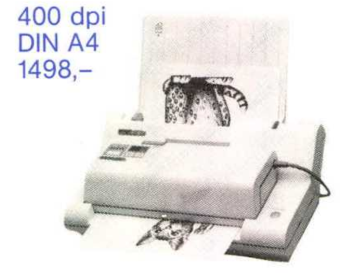
Farbscanner: (alle Scanner inkl. unten aufgeführter Software)

Charly-Color: 400 dpi Farbhandscanner, 105 mm breit, 262 144 Farben	DM 1598,–
Epson GT-6000: 600 dpi Flachbettscanner, 16 Mio. Farben, integriertes DMA-Interface (Anschluß ohne Rechnerumbau und sehr schnelles Scannen)	DM 4998,–

„C"-Programmierer gesucht

Für ein größeres Softwareprojekt suchen wir Programmierer, die als freie Mitarbeiter, in fester Anstellung oder als Praktikanten in unserem Team mitarbeiten wollen. Einfach Programmierinfo anfordern!

400 dpi
DIN A4
1498,–



Charly Page

Farbe!
400 dpi
1598,–



Charly Color

echte
400 dpi;
32 Graustufen
598,–



Charly

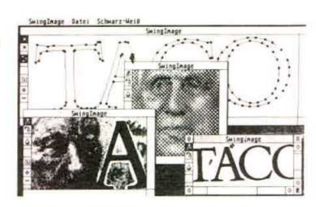


Image: Bildverarbeitung und Vektorisierung

Bildverarbeitung und Vektorisierungssoftware: (bei allen Scannern im Preis enthalten)

Image: Raster- und Vektorteil, ermöglicht Retuschierung und Fotomontage mit echten Graustufenbildern. Auch gerasterte Bilder können in echte Graustufenbilder oder Vektorgrafiken mit Bézierkurven umgerechnet werden.

Rasterteil: 2, 4, 16, 64 oder 256 Graustufen, beliebig große Bilder, 7 Fenster, flexibles Treiberkonzept, TIFF, GEM-Image, PostScript u.v.m. werden unterstützt, Farbunterstützung..

Vektorteil: Vollautomatische Vektorisierung mit Nacheditierungsmöglichkeit, Treiber für (Schneide-)Plotter und CAD-Systeme und DTP-Programme inkl. Updateservice, Hotline und allen Treibern DM 348,–

Auf alle Geräte gewähren wir 1 Jahr Garantie. Alle Preise inkl. 14 % Mehrwertsteuer. Gerne senden wir Ihnen ausführliche Unterlagen und Informationen zu unserem weiteren Produktprogramm für STs und PCs zu.

Alle Angaben ohne Gewähr. Änderungen vorbehalten.



Wilhelm Mikroelektronik · Süggelstraße 31 · 4670 Lünen · Telefon: (02306) 25299

Prozedur *NaechsterOperand()*, die feststellt, wo der nächste Operand anfängt, bzw. eine Fehlermeldung liefert, falls vor der ')' nur noch ein ',' zu finden ist. Mit diesen Hilfsprozeduren werden nun die möglichen vier Argumente "abgeklopft":

Das erste Argument wird mit *TestRegVar()* überprüft, und falls kein Register und auch keine Variable vorgefunden werden, weiterhin darauf getestet, ob es eine Zahl ist. Ansonsten setzt es eine Fehlermeldung und einen Programmabbruch! Das gleiche ist der Fall, wenn auf das erste Argument kein zweites folgt, denn wer 'A' sagt, der muß auch 'B' sagen.

Das zweite Argument kann jetzt, falls die Operation TST vorliegt, eine Boolsche Variable sein und wird dann weiter unten wie ein drittes Argument behandelt, ansonsten kann das zweite Argument nur ein Register oder eine Variable sein. *TestRegVar()* schaut nach, und falls beides nicht paßt, gibt es wieder eine Fehlermeldung.

Das dritte Argument (bzw. das zweite bei TST) kann die Boolsche Variable von CMP sein oder ein Zielregister von SINCOS. Mit dem Universalgenie *TestRegVar()* wird entweder das Zielregister identifiziert oder der Variablenname diesmal nach *fpuop.Boolean* kopiert.

Wurde eine Boolsche Variable vorgefunden, muß es auch einen Vergleichstyp geben. Mit *TestVgl()* wird er identifiziert bzw. das Fehlen festgestellt und dem Anwender mitgeteilt.

Zum Abschluß wird dann noch der *fpuop.Zeiger* hinter ein eventuelles Semikolon gesetzt. Die Analyse endet damit, daß *Syntaxpruefung()* dieselbige im Zusammenhang der gefundenen Strukturen überprüft.

Kreation der FPU-Anweisungen

Der Aufbau des Precompilats wird von der Prozedur *Codegenerierung()* des Moduls *Codierung* durchgeführt.

Bevor jedoch der Daten-Record ausgeschlachtet wird, sind noch einige Vorbereitungen zu treffen, damit die Kommentierung und das Einrücken der Zeilen korrekt ablaufen. Dies übernimmt die Prozedur *TransferZeile*, die anhand der Zeiger *fpuop.Zeiger* und fpuop.Zeiger Anfang die ursprüngliche Zeile in die Teile *AnfangZeile*, *FlOp* und *EndeZeile* zerlegt. Dann setzt man die Zeile wieder aus den Stücken *AnfangZeile*, *"(*"*, *"FlOp" und "*)"* zusammen und schreibt sie in das Ausgabe-File. Zusätzlich erfolgt die Feststellung, wie tief weitere zusätzliche Zeilen einzurücken sind. Abschließend wird die *fpuop.Zeile* gleich entsprechend vorbereitet, so daß man den neuen Text anhängen kann.

Bei der sich jetzt anschließenden Synthese der Maschinencodes wird eine Aufgliederung in verschiedene Klassen von Operationen getroffen, die jeweils von unterschiedlichen Prozeduren bearbeitet werden:

RESET, STATUS: werden wegen der Kürze sofort bearbeitet.

CMP, TST: werden von der Prozedur CmpTst übernommen.

REGISTER/REGISTER-ANWEISUNGEN: Auswertung durch *RegReg*

SPEICHER/REGISTER-ANWEISUNGEN: Auswertung durch *MemReg*

REGISTER-SPEICHER-TRANSFER: Diesen Fall übernimmt die Prozedur *RegMem*.

```
902:            sglmul : Code := Code + 39 |
903:            sub    : Code := Code + 40 |
904:            cmp    : Code := Code + 56 |
905:            tst    : Code := Code + 58 ;
906:          END ;
907:     ELSIF fpuop.Operation = sincos
908:       THEN Code := Code + 6 * 8 +
                ORD(fpuop.Argument3) - ORD(Reg0) ;
909:       ELSE StrToVal(fpuop.VarNum, romoffsetl,
                       sign, 10, error) ;
910:            romoffsets := romoffsetl ;
911:            Code := Code + romoffsets ;
912:     END ;
913:     IF NOT ( ((fpuop.Operation = move) AND
                  (fpuop.Argument2 < Reg0)) OR
914:              (fpuop.Operation = movecr) )
915:       THEN IF fpuop.Argument1 = Var
916:              THEN Code := Code +
                  ORD(fpuop.Extension) * 1024 ;
917:            ELSIF fpuop.Argument1 = Num
918:              THEN Code := Code +
                  ORD(fpuop.Extension) * 1024 ;
919:              ELSE Code := Code + ( ORD
           (fpuop.Argument1) - ORD(Reg0) ) * 1024 ;
920:            END ;
921:            Code := Code + ( ORD(fpuop.Argument2
                       ) - ORD(Reg0) ) * 128 ;
922:     ELSIF fpuop.Operation = movecr
923:       THEN Code := Code + 7 * 1024 ;
924:            Code := Code + ( ORD(fpuop.Argument2
                       ) - ORD(Reg0) ) * 128 ;
925:       ELSE Code := Code + ORD(fpuop.Extension)
                       * 1024 ;
926:            Code := Code + ( ORD(fpuop.Argument1
                       ) - ORD(Reg0) ) * 128 ;
927:     END ;
928:     RETURN Code ;
929:   END RechenCode;
930:
931:   PROCEDURE CmpTst ;
932:   BEGIN
933:     IF fpuop.Argument1 >= Reg0
934:       THEN RegReg ;
935:       ELSE MemReg ;
936:     END ;
937:     LoadAdr(fpuop.Boolean) ;
938:     Make3Code(031FCH, ORD(fpuop.Argument4),
                   0FA4EH) ;
939:       (* move.w #Condition,$ConditionReg *)
940:     Make3Code(00C78H, 00801H, 0FA40H) ;
941:       (* cmp.w #0801, $ResponseReg *)
941:     Make3Code(057C0H, 04400H, 01080H) ;
942:       (* seq.b d0 ; addq.b #1,d0 ;
            move.b d0,(a0) *)
943:   END CmpTst ;
944:
945:   PROCEDURE RegReg;
946:   BEGIN
947:     Make3Code(031FCH, RechenCode(), 0FA4AH) ;
                 (* move #Code, $CommandReg *)
948:     CASE fpuop.Operation OF
949:       abs, add, cmp, getexp, getman, int,
950:       intrz, move, movecr, neg, scale,
         sglmul, sub, tst : ResponseTest ;
951:                          Nops(0) |
952:       mod, mul, sgldiv : ResponseTest ;
953:                          Nops(1) |
954:       div, rem, sqrt : ResponseTest ;
955:                        Nops(6) |
956:       ELSE Nops(88) ;
957:     END ;
958:   END RegReg ;
959:
960:
961:   PROCEDURE MemReg ;
962:   VAR longint : LONGINT ;
963:       real : REAL ;
964:       longreal : LONGREAL ;
965:       shortcard : ARRAY [1..4] OF SHORTCARD ;
966:       adr : ADDRESS ;
967:       poishortcard : POINTER TO SHORTCARD ;
968:       error, signed : BOOLEAN ;
969:   BEGIN
970:     IF fpuop.Argument1 = Var
971:       THEN CASE fpuop.Extension OF
972:              B, W, D, E :LoadAdr(fpuop.VarNum)|
973:              L, S       :LoadData ;        →
```

```
 974:             END ;
 975:         ELSE CASE fpuop.Extension OF
 976:             B, W : StrToVal(fpuop.VarNum,
                      longint, signed, 10, error) ;
 977:                   shortcard[1] := longint |
 978:             L    : StrToVal(fpuop.VarNum,
                      longint, signed, 10, error) ;
 979:                   adr := ADR(longint) ;
 980:                   poishortcard := adr ;
 981:                   shortcard[1] :=
                              poishortcard^ ;
 982:                   poishortcard := adr + 2 ;
 983:                   shortcard[2] :=
                              poishortcard^ |
 984:             S    : StrToReal(fpuop.VarNum,
                          longreal, error) ;
 985:                   real := longreal ;
 986:                   adr := ADR(real) ;
 987:                   poishortcard := adr ;
 988:                   shortcard[1] :=
                              poishortcard^ ;
 989:                   poishortcard := adr + 2 ;
 990:                   shortcard[2] :=
                              poishortcard^ |
 991:             D    : StrToReal(fpuop.VarNum,
                          longreal, error) ;
 992:                   adr := ADR(longreal) ;
 993:                   FOR longint := 1 TO 4 DO
 994:                     poishortcard := adr ;
 995:                     shortcard[longint] :=
                                   poishortcard^ ;
 996:                     adr := adr + 2 ;
 997:                   END |
 998:             END ;
 999:       END ;
1000:       Make3Code(031FCH, RechenCode(), 0FA4AH) ;
            (* move.w #Code, $CommandReg *)
1001:       ResponseTest ;
1002:       IF fpuop.Argument1 = Var
1003:         THEN CASE fpuop.Extension OF
1004:             B : Make2Code(011D0H, 0FA50H) |
                     (* move.b (a0), $Operand *)
1005:             W : Make2Code(031D0H, 0FA50H) |
                     (* move.w (a0), $Operand *)
1006:             L, S : Make2Code(021C0H, 0FA50H) |
                     (* move.l d0, $Operand *)
1007:             D : Make4Code(021D8H, 0FA50H,
                                021D0H, 0FA50H) |
1008:                (* move.l (a0)+, $Operand ;
                        move.l (a0), $Operand *)
1009:             E : Make6Code(021D8H, 0FA50H,
                  021D8H, 0FA50H, 021D0H, 0FA50H) ;
1010:                (* 2 mal move.l (a0)+,
                $Operand ; move.l (a0), $Operand *)
1011:             END ;
1012:         ELSE CASE fpuop.Extension OF
1013:             B : Make3Code(011FCH,
                                shortcard[1],
                     0FA50H) |
1014:                (* move.b #$val, $Operand *)
1015:             W : Make3Code(031FCH,
                                shortcard[1],
                     0FA50H) |
1016:                (* move.w #$val, $Operand *)
1017:             L, S : Make4Code(021FCH,
              shortcard[1], shortcard[2], 0FA50H) |
1018:                (* move.l #$val, $Operand *)
1019:             D : Make4Code(021FCH,
                                shortcard[1],
                     shortcard[2], 0FA50H) ;
1020:                (* move.l #$valh, $Operand *)
1021:                Make4Code(021FCH,
                                shortcard[3],
                        shortcard[4], 0FA50H) ;
1022:                (* move.l #$vall, $Operand *)
1023:             END ;
1024:       END ;
1025:       ResponseTest ;
1026:       CASE fpuop.Operation OF
1027:         abs, add, cmp, getexp, getman, int,
            intrz, mod, move,
1028:         movecr, mul, neg, scale, sgldiv, sglmul,
            sub, tst : Nops(0) |
1029:         div, rem, sqrt : Nops(4) ;
1030:         ELSE Nops(88) ;
1031:       END ;
1032:     END MemReg ;                              →
```

Diese verschiedenen Prozeduren benutzen ihrerseits einen Satz gemeinsamer Hilfsprozeduren:

MAKE1CODE() BIS MAKE6CODE(): Diese Prozeduren erzeugen eine Zeile mit einem bis sechs Hexadezimalwörtern. Da sie sich alle auf eine gemeinsame Prozedur *MakeCode()* stützen, für die sie nur eine vereinfachte Datenübergabe bewirken, läßt sich die Art des für die Hexadezimalwörter erzeugten Strings recht einfach manipulieren.

LOADADR(): erzeugt einen String, um eine Variablenadresse in das Adreßregister a0 zu lesen. Dies kann entweder die Adresse eines Operanden sein oder die der Boolschen Variablen.

LOADDATA: Dient dem direkten Einlesen einer Variablen in das Datenregister d0.

NOPS(): Hiermit wird eine gewünschte Anzahl von Nops für den Hauptprozessor erzeugt, damit sich dieser mit der FPU synchronisieren kann. Übergibt man der Prozedur eine Zahl größer sechs, produziert sie eine Test- und Warteschleife, die den Hauptprozessor so lange das Response-Register lesen läßt, bis dieses signalisiert, daß die FPU fertig ist.

RESPONSETEST: Da das Responseregister sehr häufig gelesen werden muß, um das Protokoll einzuhalten, wurde diese kurze Prozedur geschaffen, um nicht immer mit Hexcodes um sich werfen zu müssen. Mit diesen Erläuterungen sollte die Codesynthese eigentlich verständlich geworden sein. Für nähere Informationen zum erzeugten Maschinencode sei auf [2], [3], [4] und [5] verwiesen.

Nach der Synthese wird der mit TransferZeile aufgehobene Zeilenrest nach fpuop.Zeile übertragen und die Zeiger entsprechend gesetzt, damit an diesem Punkt eine erneute Analyse einsetzen kann. Damit schließt die Prozedur Codierung und auch die Beschreibung der Arbeitsweise des Precompilers ab.

Zukunftsmusik

Daß ein Werkzeug für Programmierer nie fertig ist, ist bekannt. Einige Punkte, die Anlaß zu weiterer Arbeit geben, sollen hier aufgeführt werden.

Da wäre zunächst die Anpassung des Precompilats an Fließkommaprozessoren, die von einem anderen als einem 16 MHz-Quarz getrieben werden. Bei einer 12 MHz-FPU wird man derzeit außer Protokollverletzungen recht wenig ernten. Wer eine 20 MHz-FPU sein Eigentum nennen kann, der wird bei allen Grundfunktionen keine Geschwindigkeitssteigerung bekommen. In beiden Fällen sind Experimente angesagt, um die nötige Anzahl an Nops zu ermitteln. Die in dieser Programmversion verwendeten Werte stammen ausschließlich aus solchen Experimenten, da die bei [3] vorgefundenen Werte zu unnötigen Verzögerungen führten und dort auch nicht alle Möglichkeiten der FPU angewandt wurden. Für die 12 MHz-FPU gibt es auch eine Holzhammermethode. Man braucht lediglich am Anfang der Prozedur *Nops() die Bedingungen IF AnzNops <= 6* durch *IF AnzNops < 0* zu ersetzen. Dies hat zur Folge, daß immer eine Test- und Warteschleife erzeugt wird, so daß die FPU beliebig langsam sein kann. Damit erzielt man natürlich keine optimalen Ergebnisse, aber man ist immer noch schneller, als wenn man ohne Precompiler arbeiten muß. Da ich weder einen 12 MHz- noch einen 20 MHz-Coprozessor in der Schublade habe, wäre ich recht dankbar, wenn jemand diese Werte bestimmen könnte.

Man kann diese Experimente am einfachsten durch Änderungen an einem vom Precompiler behandelten Text machen. Ich habe denselben Weg eingeschlagen und Änderungen erst dann in den Precompiler übernommen, wenn sie durch die Experimente abgesichert waren. Eine andere wunde Stelle ist die Tatsache, daß das Hänisch-Modula trotz seiner überragenden Eigenschaften (noch) keine weite Verbreitung erfahren hat und das Arbeiten mit einem Public Domain-Compiler auch nicht jedermanns Sache ist, da der fehlende Support oft zum Flaschenhals ganzer Projekte werden kann. Die Anpassung an den TDI-Compiler ist hier ein erster Schritt.

Wer mir die nötigen Informationen für seinen Compiler beschaffen kann oder ihn mir, da er ihn geschrieben hat, vielleicht sogar ausleihen kann, darf sicher sein, daß ich mein Möglichstes versuchen werde. Der Compiler muß folgende Eigenschaften haben:

Die Register a0 und d0 müssen frei verfügbar sein.

Man muß ohne Verrenkungen eine Adresse nach a0 und eine 32 Bit-Variable nach d0 laden können.

Es muß eine Anweisung geben, mit der man Hexcodes in den Modulatext integrieren kann, so daß diese direkt ausgeführt werden. Ersatzweise wäre auch ein Inline-Assembler denkbar, sofern zwei voneinander abgeschlossene, aber direkt aufeinander folgende Assembler-Sequenzen auch direkt hintereinander ausgeführt werden.

Die Sprache muß Kommentare erlauben, die wie in Modula oder C in Klammern eingeschlossen sind. Die Art der Klammern ist beliebig.

Falls die obigen fünf Bedingungen wider mein Wissen noch in einer anderen Sprache als Modula erfüllt sind, kann der Precompiler auch an diese Sprache angepaßt werden. Lediglich bei ADA würde ich mich weigern, da ich Leute, die - für wen auch immer - in dieser SDI-Sprache programmieren, nicht auch noch unterstützen will. Eine Anpassung an ADA stellt einen Copyright-Verstoß dar. Eine Weitergabe wird entsprechend geahndet.

Wer Anpassungen selber erfolgreich durchgeführt hat, möge mir das geänderte Programm bitte zuschicken, damit ich die diversen Varianten nach Möglichkeit zu einem Programm vereinigen kann. Über Anregungen und Anwendungsberichte würde ich mich sehr freuen.

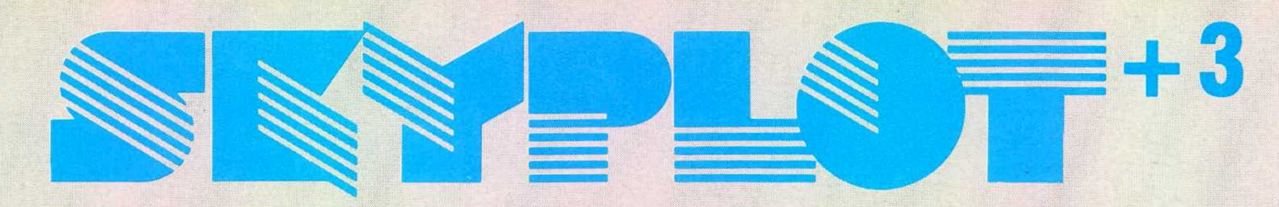
Umsetzung in die Praxis

Am Beispiel einer Integralberechnung nach der 3/8-Regel soll der Weg von einem normalen Programm zu einem mit dem Precompiler optimierten gezeigt werden. Die Quelltexte hierzu finden sich in den Listings 2 bis 4.

Will man nicht alles optimieren, weil es einem zuviel Aufwand bedeutet, sollte man sich zunächst die Schleifen aussuchen und bei den am tiefsten liegenden Teilen anfangen. Hat man keine ergiebigen Schleifen, sollte man sich Prozeduren vornehmen, die häufiger aufgerufen werden. Zu beachten ist, daß bei nichtoptimierten Rechenteilen die Register fp0 und fp1 benutzt werden. Programmiert man von "innen" nach "außen", ergeben sich dadurch keine Probleme, ansonsten sollte man fp0 und fp1 nur für temporäre Werte benutzen, bzw. sichern. Bei interruptgesteuerten bzw. pseudoparallelen Programmen sollte man ebenfalls daran denken, die Register, die man ändern wird, vorher zu sichern und hernach wieder in den alten Zustand zu versetzen,

```
1033:
1034:      PROCEDURE RegMem ;
1035:      BEGIN
1036:        LoadAdr(fpuop.VarNum) ;
1037:        Make3Code(031FCH, RechenCode(), 0FA4AH) ;
                      (* move #Code,$Command *)
1038:        ResponseTest ;
1039:        CASE fpuop.Extension OF
1040:          B, W, L : Nops(4) |
1041:          S, D : Nops(2) |
1042:          E : Nops(0) ;
1043:        END ;
1044:        ResponseTest ;
1045:        CASE fpuop.Extension OF
1046:          B : Make2Code(010B8H, 0FA50H) |
                  (* move.b #$fa50, (a0) *)
1047:          W : Make2Code(030B8H, 0FA50H) |
                  (* move.w #$fa50, (a0) *)
1048:          L, S : Make2Code(020B8H, 0FA50H) |
                  (* move.l #$fa50, (a0) *)
1049:          D : Make4Code(020F8H, 0FA50H, 020B8H,
                        0FA50H) |
1050:             (* move.l #$fa50, (a0)+ ;
                     move.l #$fa50, (a0) *)
1051:          E : Make6Code(020F8H, 0FA50H, 020F8H,
                        0FA50H, 020B8H, 0FA50H) ;
1052:             (* 2 mal move.l #$fa50, (a0)+ ;
                     move.l #$fa50, (a0) *)
1053:        END ;
1054:        ResponseTest ;
1055:      END RegMem ;
1056:
1057:    BEGIN
1058:      TransferZeile ;
1059:      IF fpuop.Operation = reset
1060:        THEN Make2Code(04278H, 0FA42H);
             (* Coprozessor-Reset *)
1061:      ELSIF fpuop.Operation = status
1062:        THEN fpuop.Extension := L ; (* wichtig für
                            das Modula der TU München *)
1063:             LoadAdr("fpustatus") ;
1064:             Make2Code(030B8H, 0FA40H);
                  (* Status-Abfrage *)
1065:      ELSIF fpuop.Operation = cmp
1066:        THEN CmpTst ;
1067:      ELSIF fpuop.Operation = tst
1068:        THEN fpuop.Argument2 := Reg0 ;
1069:             CmpTst ;
1070:      ELSIF ((fpuop.Argument1 >= Reg0) AND
                (fpuop.Argument2 >= Reg0)) OR
1071:            (fpuop.Operation = movecr)
1072:        THEN RegReg ;
1073:      ELSIF (fpuop.Argument1 < Reg0) AND
                (fpuop.Argument2 >= Reg0)
1074:        THEN MemReg ;
1075:        ELSE RegMem ;
1076:      END ;
1077:      Copy(fpuop.Zeile, Einrueckstring, 0,
               Einrueckung) ;
1078:      INC(fpuop.Zeilennummer) ;
1079:      Concat(fpuop.Zeile, EndeZeile) ;
1080:      fpuop.Zeiger := Einrueckung ;
1081:      fpuop.ZeigerAnfang := fpuop.Zeiger ;
1082:    END Codegenerierung ;
1083:  END Codierung ;
1084:
1085:  PROCEDURE Compilieren ;
1086:  VAR Resultat1, Resultat2, Resultat3 : BOOLEAN ;
1087:      ch        : CHAR ;
1088:      Abschnitt: ARRAY [0..10] OF CHAR ;
1089:  BEGIN
1090:    ParserInit ;
1091:    ReadLine(fpuop.Zeile) ;
1092:    IF RDone
1093:      THEN INC(fpuop.Zeilennummer) ;
1094:           fpuop.Zeiger := 0 ;
1095:    END ;
1096:    IF NOT Compileridentifizieren()
1097:      THEN RETURN ;
1098:    END ;
1099:    REPEAT
1100:      fpuop.Operation := noop ;
1101:      fpuop.Argument1 := nouebg ;
1102:      fpuop.Argument2 := nouebg ;
1103:      fpuop.Argument3 := nouebg ;
1104:      fpuop.Argument4 := novgl ;
1105:      fpuop.VarNum := "" ;
```

→



Ein paar Worte zum Wahnsinn:

Um es gleich deutlich zu machen: Es geht hier um das Programm **SKYPLOT**, genauer um die neueste Version **SKYPLOT PLUS 3**.

Es gab einmal einen „absoluten Wahnsinn" in Form von **SKYPLOT PLUS**, der dann dem „gesteigerten Wahnsinn" in Gestalt des Nachfolgers **SKYPLOT PLUS 2** weichen mußte. Nun fällt uns leider keine weitere Steigerung mehr zum Wahnsinn ein, wo **SKYPLOT PLUS 3** diese doch verdient hätte! Was tun?

Wir lassen also die kühlen Fakten für sich sprechen, ganz vernünftig:

„...the most sophisticated astronomical simulation package that we have ever seen on a microcomputer"
(_The_Planetarian_)

„...die Möglichkeiten sind selbst bei häufiger Benutzung kaum auszuschöpfen"
(c't)

„...gäbe es einen Oscar oder eine goldene Palme für Atari-Programme, wäre SKYPLOT ein Kandidat dafür... ein echter Grund, sich einen Atari ST zuzulegen"
(XEST)

„...ein unglaubliches Programm... einfacher und eindrucksvoller läßt sich einem Interessierten der Kosmos kaum näherbringen"
(ATARI Magazin)

„...goldenes Byte für SKYPLOT"
(Computer persönlich)

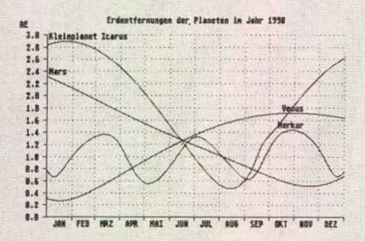
„...von dem Programm sehr begeistert... erwies sich auch im professionellen Gebrauch sehr nützlich"
(Max-Planck-Institut für Radioastronomie Bonn)

„...if you own an Atari and enjoy astronomy, you must get this program"
(Sky & Telescope)

Nun genug der Meinungen, schließlich beziehen sich die Pressestimmen ja alle auf die *alten* Versionen, denen zu **SKYPLOT PLUS 3** mindestens über ein Jahr an Entwicklungszeit fehlen!

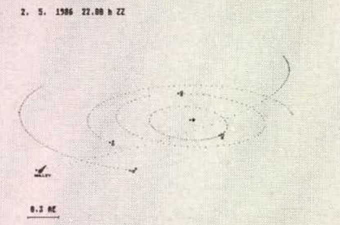
Irgendwo muß diese Arbeit wohl stecken, und zwar hier:

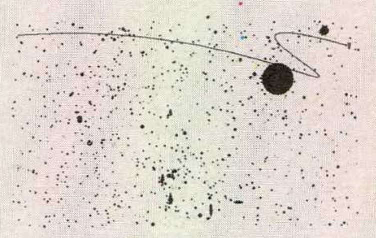
- bis zu 64000 Sterne
- bis zu 32000 Nebel etc.
- umfangreiche On-Line-Hilfen
- Echtzeitsimulation
- jede Mange Diagramme und Simulationen zur Verdeutlichung von astronomischen Sachverhalten
- Bilder laden oder speichern: IMG, AIM, STAD, Neochrome, Degas (Elite), komprimiert



- eingebaute DOS-Funktoinen: Dateien löschen, Programme starten, Formatieren
- umfangreicher Parameter für Grafik und Drucker
- Zeichnen, Speichern und Laden von Bewegungsbahnen
- Plotausgabe, auch für HPGL-kompatible Plotter
- Laden von Daten stark beschleunigt

- Finsterniskanon
- Benutzung der Hardwareuhr
- Zonenzeit und Zeitzonen
- direkte Steuerung von Vergrößerung, Ausschnitt etc. durch Tasten
- 3D-Darstellung, auch für Stereoskope
- Grafiken bearbeiten: Spiegeln, Beschriften etc.
- Speichern von Sequenzen





- eingebaute Editoren für Textdateien und Sternbildhilfslinien
- Datenausgabe auf Drucker: Ephemeriden, Kalender etc.
- jede Menge Hardcopies eingebaut, auch für 24- oder 48-Nadeldrucker
- mitgelieferter Bitmaptreiber erreicht Auflösungen von 6912★4320 Pixeln oder mehr

Schluß mit dem Wahnsinn — her mit SKYPLOT PLUS 3!

* alle Preise sind unverbindlich empfohlene Verkaufspreise

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir

____St. Skyplot plus 3 DM 198.–
____St. Skyplot plus 3 (Co-Prozessor Version) DM 248.–
____St. Up-Date auf Skyplot plus 3 ohne Modul DM 50.–
____St. Up-Date (von Skyplot plus 3 auf Co-Prozessor Version) mit Modul DM 80.–
____St. Up-Date (von Skyplot plus 3 auf Co-Prozessor Version) ohne Modul DM 50.–
____St. Up-Date von älteren Versionen jeweils Mehrpreis von DM 50.–
____St. Up-Date auf Skyplot plus 3 mit Modul DM 80.–

zuzügl. Versandkosten DM 6,– (Ausland DM 10,– unabhängig von der bestellten Stückzahl

Heim Verlag

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte

damit es nicht zu Komplikationen kommt. Ist man sich nicht sicher, daß man mit den acht Registern auskommt, sollte man bei der Registervergabe ebenfalls erst die zeitintensivsten Teile bedenken, denn je weniger man Werte zwischen dem Hauptspeicher und dem Coprozessor verschieben muß, desto eher ist der Computer fertig. In unserem Beispielprogramm wäre der rechenintensivste Teil also die Schleife über k, die je nach gewünschtem N einige tausend Mal durchlaufen wird. Man sollte seine Gedanken also zunächst auf diesen Teil konzentrieren und sich überlegen, welche Werte man in Registern haben muß, damit sie nicht in dieser Schleife geladen werden müssen. Dies sind a und h. Des weiteren benötigt man noch je ein Register für die dreifach bzw. die doppelt gewichtete Summe. Da man bei $sin(x)/x$ das x zweimal benötigt, braucht man noch ein weiteres Register, damit man dieses x nicht zerstören muß. Damit kommt man auf insgesamt fünf Register. Damit bleiben noch drei übrig. Man kann also auch die Variable b aufheben, die man am Anfang zur Berechnung von h und am Schluß noch für eine Funktionswertberechnung benötigt. Damit sind also alle Variablen in einem Register aufgehoben, was der Idealfall ist. Diese Überlegungen sollte man in freier Form irgendwie schriftlich festhalten, wobei sich der Computer hier eventuell anbietet, da man mit ihm schnell "radieren" kann, ohne daß alles im Chaos endet. Wie ein solches Konzept aussehen kann, zeigt Listing 3.

Ist in dem Konzeptpapier der Algorithmus gereift, kann man ihn schließlich in Precompiler-Anweisungen umsetzen und testen. Man sollte dabei nicht zu sehr mit Kommentierungen sparen. Da der Precompiler bewußt in das Ausgangs-File zurückschreibt, sollte man sich von dem Modulatext ein Duplikat anlegen, das man notfalls wieder zum Haupt-File machen kann. Innerhalb der Entwicklungsumgebung von Hänisch-Modula reicht es, nach dem Aufruf von Precompiler und Compiler den Text, der sich unverändert im Editor befindet, einfach ein zweites Mal abzuspeichern. Damit existieren dann immer nur die Stammversion des Programms und die Übersetzungen des Precompilats.

Rechts noch ein Ausschnitt aus einem Travelling-Salesman-Programm, in dem die Berechnung von vier Hypothenusen zu optimieren war, die in dem Programm etwa 200000mal durchlaufen wird.

Benchmarks

Um in etwa den Zeitgewinn abschätzen zu können, wurde das obige Programm zur Integralberechnung als Benchmark herangezogen. Die erhaltenen Werte dürfen allerdings nicht kritiklos gelesen werden. Zunächst bestehen echte Programme nicht nur aus Mathematik und zum anderen werden in der Regel Zugriffe auf Datenfelder nötig sein, die den Vorsprung eines Coprozessors bzw. des Precompilers etwas senken. Je höher der Anteil an trigonometrischen Funktionen in einem Programm ist, desto geringer wird der Vorsprung der optimierten Programme sein, da hier die für die Berechnung benötigte Zeit höher ist als die für den Datentransfer. Dies ist allerdings in den wenigsten Programmen der Fall. In dem Travelling-Salesman-Programm, das sich auch auf der Programmdiskette befindet, ist die Berechnung der Hypothenusen, die optimiert wurde, mit einer umfangreichen Sortierung, Dateizugriffen und Grafikausgabe verbunden, und es wurde trotzdem eine Verbesserung um 40% erzielt. In der Regel wird der Zeitgewinn wohl zwischen den Extremen Integralberechnung und Travelling-Salesman liegen.

Zu den Programmversionen, die ohne Coprozessor laufen, ist zu sagen, daß das PD-Modula keine LONGREAL-Zahlen unterstützte, und daß TDI-Modula zwar das Datenformat LONGREAL anbietet, jedoch in Wirklichkeit nur mit einer Mantisse von 20 Bits rechnet, also sogar mit einer geringeren Genauigkeit als das PD-Modula im REAL-Format (24 Bit-Mantisse) (siehe Tabelle 2).

Ich hoffe, mit diesen beiden Artikeln möglichst vielen Programmierern die FPU etwas näher und die Computer ins Schwitzen gebracht zu haben.

Jens Decker

```
PROCEDURE Hypot(stadt, stadt1: staedte) : LONGREAL ;
BEGIN
(* Berechnet den Abstand zweier Städte. *)
  RETURN Sqrt((xkoord[stadt]-xkoord[stadt1])*
         (xkoord[stadt]-xkoord[stadt1])+
         (ykoord[stadt]-ykoord[stadt1])*
         (ykoord[stadt]-ykoord[stadt1]));
END Hypot ;

   ELSE (* neuwert:=momwert+Hypot
         (momzustand[zeiger],momzustand
         [vorg(zeiger1)])+Hypot(momzustand
         [nachf(zeiger)],momzustand[zeiger1]) -
Hypot(momzustand[zeiger],momzustand
         [nachf(zeiger)])-Hypot(momzustand
         [zeiger1], momzustand[vorg(zeiger1)]);
         *) (* Normaler Weg *)
    b:=momzustand[vorg(zeiger1)] ;
    c:=momzustand[nachf(zeiger)] ;

    Fmove(xkoord[momzustand[zeiger]],fp0) ;
    Fmove(fp0,fp4) ;
    Fmove(ykoord[momzustand[zeiger]],fp1) ;
    Fmove(fp1,fp5) ;
    Fmove(xkoord[b],fp2) ;
    Fmove(ykoord[b],fp3) ;
    Fsub(fp2,fp0) ;
    Fsub(fp3,fp1) ;
    Fmul(fp0,fp0) ;
    Fmul(fp1,fp1) ;
    Fadd(fp1,fp0) ;
    Fsqrt(fp0,fp7) ;
    (*Hypot(momzustand[z],momzustand[vorg(z1)])*)
    Fadd(momwert,fp7) ;
    Fmove(xkoord[c],fp0) ;
    Fmove(ykoord[c],fp1) ;
    Fsub(fp0,fp4) ;
    Fsub(fp1,fp5) ;
    Fmul(fp4,fp4) ;
    Fmul(fp5,fp5) ;
    Fadd(fp5,fp4) ;
    Fsqrt(fp4,fp4) ;
    (*Hypot(momzustand[z],momzustand[nachf(z)])*)
    Fsub(fp4,fp7) ;

    Fmove(xkoord[momzustand[zeiger1]],fp4) ;
    Fmove(ykoord[momzustand[zeiger1]],fp5) ;
    Fsub(fp4,fp0) ;
    Fsub(fp5,fp1) ;
    Fmul(fp0,fp0) ;
    Fmul(fp1,fp1) ;
    Fadd(fp1,fp0) ;
    Fsqrt(fp0,fp0) ;
    (*Hypot(momzustand[nachf(z)],momzustand[z1])*)
    Fadd(fp0,fp7) ;

    Fsub(fp2,fp4) ;
    Fsub(fp3,fp5) ;
    Fmul(fp4,fp4) ;
    Fmul(fp5,fp5) ;
    Fadd(fp5,fp4) ;
    Fsqrt(fp4,fp4) ;
    (*Hypot(momzustand[z1],momzustand[vorg(z1)])*)
    Fsub(fp4,fp7) ;
    Fmove(fp7,neuwert) ;
```

Abbildung 1: Ein Ausschnitt aus einem Travelling-Salesman-Programm

```
1106:      fpuop.Boolean := "" ;
1107:      Resultat1 := Fsuchen() ;
1108:      IF Resultat1
1109:       THEN Resultat2 := Operationidentifizie-
                                          ren();
1110:         IF Resultat2 AND (fpuop.Operation #
                                          reset)
1111:            AND (fpuop.Operation # status)
1112:              THEN IF Extensionidentifizieren()
1113:                THEN IF SucheOperand1()
1114:                  THEN IF OperandenIdentifizie-
ren()
1115:                    THEN IF Syntaxpruefung()
1116:                      THEN Resultat3 :=
                                Codegenerierung() ;
1117:                      END ;
1118:                    END ;
1119:                  END ;
1120:                END ;
1121:              ELSE IF (fpuop.Operation = reset) OR
                          (fpuop.Operation = status)
1122:                      THEN SucheEnde ;
1123:                           Resultat3 :=
                                    Codegenerierung() ;
1124:            END ;
1125:          END ;
1126:      END ;
1127:   (*    IF Resultat1 AND Resultat2 AND RDone
1128:        THEN InOut.WriteString(fpuop.Zeile) ;
                  InOut.WriteLn ;
1129:             InOut.WriteString("Operation: ") ;
                  InOut.WriteString(Operationen[ORD
                  (fpuop.Operation)]) ; InOut.WriteLn ;
1130:             InOut.WriteString("Extension: ") ;
                  InOut.WriteString(Extensions[ORD
                  (fpuop.Extension)]) ; InOut.WriteLn ;
1131:             InOut.WriteString("Argument1: ") ;
                  InOut.WriteString(Operanden[ORD
                  (fpuop.Argument1)]) ; InOut.WriteLn ;
1132:             InOut.WriteString("Argument2: ") ;
                  InOut.WriteString(Operanden[ORD
                  (fpuop.Argument2)]) ; InOut.WriteLn ;
1133:             InOut.WriteString("Argument3: ") ;
                  InOut.WriteString(Operanden[ORD
                  (fpuop.Argument3)]) ; InOut.WriteLn ;
1134:             InOut.WriteString("Argument4: ") ;
                  InOut.WriteString(Vergleiche[ORD
                  (fpuop.Argument4)]) ; InOut.WriteLn ;
1135:             InOut.WriteString("VarNum    : ") ;
                  InOut.WriteString(fpuop.VarNum) ;
                  InOut.WriteLn ;
1136:             InOut.WriteString("Boolean   : ") ;
                  InOut.WriteString(fpuop.Boolean) ;
                  InOut.WriteLn ;
1137:             InOut.WriteLn ;
1138:             InOut.Read(ch) ;
1139:        END ; *)
1140:    UNTIL (NOT (Resultat1 AND RDone)) ;
1141:  END Compilieren ;
1142:
1143:  BEGIN
1144:    ApID := ApplInit() ;
1145:    IF ApID > -1
1146:      THEN Intro ;
1147:           IF OeffneStreams()
1148:             THEN Compilieren ;
1149:               (*  InOut.WriteString(CompIDs
                  [Compiler]) ; InOut.WriteLn ; *)
1150:                 SchliesseStreams ;
1151:           END ;
1152:           ApplExit ;
1153:    END ;
1154:  END FPUPreprozessor.
```

Listing 1: Der zweite Teil des Precompilers.

```
 1:  MODULE Int2pd ; (* Für PD-Modula der TU München *)
 2:
 3:  FROM RealInOut IMPORT ReadReal, WriteReal ;
 4:  FROM InOut IMPORT Read, ReadInt, WriteString,
                           WriteLn ;
 5:  FROM MathLib0 IMPORT sin, real ;
 6:
 7:  VAR a, b, integral : REAL ;
 8:      h, x, sum3, sum2 : REAL ;
 9:      k, N : INTEGER ;
10:      ch : CHAR ;
11:  BEGIN
12:    WriteString("Dies ist ein Programm zur
                  Berechnung von     ") ; WriteLn ;
13:    WriteString("Integralen nach der 3/8-Formel.
                    ") ; WriteLn ;
14:
15:    WriteString("Näheres z.B. bei G.Engeln-Müllges/
                    F.Reutter:") ; WriteLn ;
16:    WriteString("Formelsammlung zur Numerischen
                    Mathematik    ") ; WriteLn ;
17:    WriteString("Mit C-Programmen, BI
                 Wissenschaftsverlag,    ") ; WriteLn ;
18:    WriteString("Mannheim, 1987 ") ; WriteLn ;
19:
20:    WriteString("Es wird die Funktion
                    ") ; WriteLn ;
21:
22:    WriteString("sin(x)/x
                    ") ; WriteLn ;
23:
24:    WriteString("integriert. Bitte geben Sie die
                    Interval-    ") ; WriteLn ;
25:    WriteString("grenzen und N/3, ein Drittel der
                    ") ; WriteLn ;
26:    WriteString("effektiven Stuetzpunkte ein:
                    ") ; WriteLn ;
27:    WriteString("[a,b] a:") ;
28:    ReadReal(a) ; WriteLn ;
29:    WriteString("[a,b] b:") ;
30:    ReadReal(b) ; WriteLn ;
31:    WriteString("N/3:") ;
32:    ReadInt(N) ;
33:    h := (b-a)/real(3*N) ;
34:    sum2 := 0.0 ;
35:    sum3 := 0.0 ;
36:    x := a ;
37:
38:    FOR k := 1 TO N-1 DO
39:      x := x + h ;
40:      sum3 := sum3 + sin(x)/x ;
41:      x := x + h ;
42:      sum3 := sum3 + sin(x)/x ;
43:      x := x + h ;
44:      sum2 := sum2 + sin(x)/x ;
45:    END ;
46:    x := x + h ;
47:    sum3 := sum3 + sin(x)/x ;
48:    x := x + h ;
49:    sum3 := sum3 + sin(x)/x ;
50:
51:    integral := 0.375 * h * (sin(a)/a + sin(b)/b
                   + 3.0 * sum3 + 2.0 * sum2) ;
52:
53:    WriteReal(integral, 12) ; WriteLn ;
54:    Read(ch) ;
55:  END Int2pd.
```

Listing 2: Ursprüngliches Programm

	Rechenzeit		
Programm	ohne	FPU	optimiert
Hänisch: sin(x)/x	2200	96	15
SPC	1320	-	15
TDI	1800	-	15
PD	840	-	14
Hänisch: x^2	220	68	6
SPC	165	-	6
TDI	150	-	6
PD	90	-	6

Tabelle 2: Die Benchmarks im einzelnen

```
a -> fp6 (* bleiben dort *)
b -> fp7 (* bleiben dort *)

N   -> fp4 (* später die dreifach gewichtete Summe *)
3   * fp4
fp7 -> fp5
fp6 - fp5
fp4 / fp5 (* h := b-a/3N nach fp5 *)

C0.0 -> fp4 (* dreifach gewichtete Summen in fp4 auf
                                                 0.0 *)
C0.0 -> fp5 (* doppelt gewichtete Summen in fp3 auf
                                                 0.0 *)

fp6, fp1 (* in fp1 wird x an f(x) uebergeben *)

FOR k := 1 TO N-1 DO
  fp5 + fp1
  sin fp1 -> fp0
  fp1 / fp0
  fp0 + fp4

  Das gleich nochmal

  Das gleiche nochmal, aber nach fp3 addieren
END ;

Nochmal zweimal das gl iche fp4

2 * fp3 (* Gewichtung der Summen *)
3 * fp4

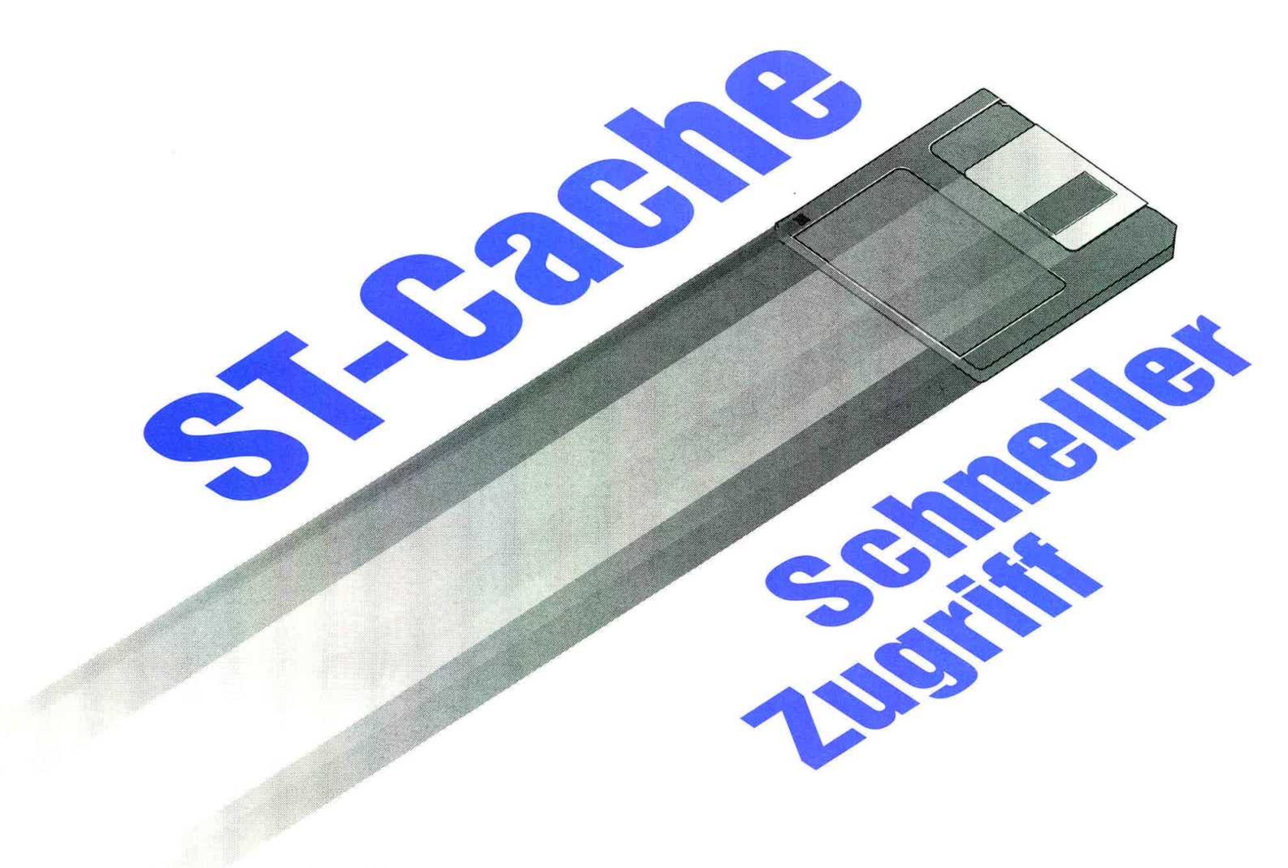
fp3, fp4 (* Summen addieren *)

Funktion für fp7 nach fp0
fp0 + fp4
Funktion für fp6 nach fp0
fp0 + fp4
fp5 * fp4 (* mit h multiplizieren *)
0.375 * fp4 (* mit 3/8 multiplizieren *)
fp4 -> integral
```

Listing 3: Konzept der Optimierung

```
 1:  MODULE IntPd ; (* Für PD-Modula der TU München *)
 2:
 3:  FROM SYSTEM IMPORT ADR, INLINE, SETREG, REG ;
 4:  FROM RealInOut IMPORT ReadReal, WriteReal ;
 5:  FROM InOut IMPORT Read, ReadInt, WriteInt,
                      WriteString, WriteLn ;
 6:
 7:  VAR a, b, integral : REAL ;
 8:      N, k, wdhl : INTEGER ;
 9:      ch : CHAR ;
10:      fpustatus : INTEGER ;
11:  BEGIN
12:    WriteString("Dies ist ein Programm zur
                    Berechnung von    ") ; WriteLn ;
13:    WriteString("Integralen nach der 3/8-Formel.
                  ") ; WriteLn ;
14:
15:    WriteString("Näheres z.B. bei G.Engeln-Müllges/
                    F.Reutter:") ; WriteLn ;
16:    WriteString("Formelsammlung zur Numerischen
                    Mathematik    ") ; WriteLn ;
17:    WriteString("Mit C-Programmen, BI
                 Wissenschaftsverlag,     ") ; WriteLn ;
18:    WriteString("Mannheim, 1987          ") ; WriteLn ;
19:
20:    WriteString("Es wird die Funktion") ; WriteLn ;
21:
22:    WriteString("sin(x)/x                ") ; WriteLn ;
23:
24:    WriteString("integriert. Bitte geben Sie die
                    Interval-   ") ; WriteLn ;
25:    WriteString("grenzen und die N/3, ein Drittel
                    der        ") ; WriteLn ;
26:    WriteString("effektiven Stützpunkte ein:
                  ") ; WriteLn ;
27:    WriteString("[a,b] a:") ;
28:    ReadReal(a) ; WriteLn ;
29:    WriteString("[a,b] b:") ;
30:    ReadReal(b) ; WriteLn ;
31:    WriteString("Die Berechnung läuft zur Zeitnahme
                    10 mal    ") ; WriteLn ;
32:    WriteString("N/3 :") ;
33:    ReadInt(N) ; WriteLn ;
34:    Freset ;
35:
36:    FOR wdhl := 1 TO 10 DO
37:      FmoveS(a, fp6) ; (* a nach fp6 *)
                    (* NUR EINFACH GENAU !!! *)
38:      FmoveS(b, fp7) ; (* b nach fp7 *)
39:
40:      FmoveW(N, fp4) ; (* NUR WORD, NICHT LONGWORD
                            WEGEN PD-MODULA !!! *)
41:      FmulB(3, fp4) ;
42:      Fmove(fp7, fp5) ;
43:      Fsub(fp6, fp5) ;
44:      FdivL(fp4, fp5) ; (* h := b-a/3N nach fp5 *)
45:
46:      Fmovecr(15, fp4) ; (* dreifach gewichtete
                              Summen in fp4 auf 0.0 *)
47:      Fmovecr(15, fp3) ; (* doppelt gewichtete
                              Summen in fp3 auf 0.0 *)
48:
49:      Fmove(fp6, fp1) ; (* in fp1 soll immer x an
                             f(x) übergeben werden *)
50:
51:      FOR k := 1 TO N-1 DO
52:        Fadd(fp5, fp1) ; (* h zu a addieren *)
53:        Fsin(fp1, fp0) ;
54:        Fdiv(fp1, fp0) ; (* f(x) = sin(x)/x *)
55:        Fadd(fp0, fp4) ; (* f(a+(3k-2)h) *)
56:
57:        Fadd(fp5, fp1) ; (* h zu a addieren *)
58:        Fsin(fp1, fp0) ;
59:        Fdiv(fp1, fp0) ; (* f(x) = sin(x)/x *)
60:        Fadd(fp0, fp4) ; (* f(a+(3k-1)h) *)
61:
62:        Fadd(fp5, fp1) ; (* h zu a addieren *)
63:        Fsin(fp1, fp0) ;
64:        Fdiv(fp1, fp0) ; (* f(x) = sin(x)/x *)
65:        Fadd(fp0, fp3) ; (* f(a+3kh) *)
66:      END ;
67:      Fadd(fp5, fp1) ; (* h zu a addieren *)
68:      Fsin(fp1, fp0) ;
69:      Fdiv(fp1, fp0) ; (* f(x) = sin(x)/x *)
70:      Fadd(fp0, fp4) ; (* f(a+(3k-2)h) *)
71:
72:      Fadd(fp5, fp1) ; (* h zu a addieren *)
73:      Fsin(fp1, fp0) ;
74:      Fdiv(fp1, fp0) ; (* f(x) = sin(x)/x *)
75:      Fadd(fp0, fp4) ; (* f(a+(3k-1)h) *)
76:
77:      FmulB(2, fp3) ; (* Summe zweifach gewichten *)
78:      FmulB(3, fp4) ; (* Summe dreifach gewichten *)
79:      Fadd(fp3, fp4) ; (* Summen addieren *)
80:      Fmove(fp7, fp1) ;
81:      Fsin(fp1, fp0) ;
82:      Fdiv(fp1, fp0) ;
83:      Fadd(fp0, fp4) ; (* f(b) zur Summe addieren *)
84:      Fmove(fp6, fp1) ;
85:      Fsin(fp1, fp0) ;
86:      Fdiv(fp1, fp0) ;
87:      Fadd(fp0, fp4) ; (* f(a) zur Summe addieren *)
88:      Fmul(fp5, fp4) ;
89:      FmulD(0.375, fp4) ; (* Integral = 3h/8 * Summe
90:                             *)
91:      FmoveS(fp4, integral) ; (* NUR EINFACH GENAU
                                  WEGEN PD-MODULA !!! *)
92:
93:      Fstatus ;
94:    END ;
95:    WriteReal(integral, 12) ; WriteLn ;
96:    WriteString("fpustatus: ") ;
       WriteInt(fpustatus, 6) ; WriteLn ;
97:    Read(ch) ;
98:  END IntPd.
```

Listing 4: Optimiertes Programm



© M.V. ZIMMERMANN

Als Alternative, aber auch als Ergänzung zu einer RAM-Disk kann dieses Accessory eine enorme Geschwindigkeitssteigerung bei Disketten[Lese]zugriffen bewirken. Dabei wird im günstigsten Fall die Geschwindigkeit einer RAM-Disk erreicht.

Das Programm aktualisiert bei jedem Diskettenzugriff eine Statistik, in der die Anzahl der Zugriffe auf jeden einzelnen Sektor der Diskette gespeichert wird, und behält dann die am häufigsten benutzten Sektoren direkt im Speicher.

Wenn jetzt ein Lesezugriff auf diese Sektoren erfolgt, werden sie nicht mehr von Diskette, sondern direkt aus dem Speicher geladen, was die ganze Sache natürlich enorm beschleunigt.

Zur Bedienung

Die Voreinstellung des Accessorys sieht folgendermaßen aus:

- Größe des Cache-Speichers: 128 KByte
- Cache-Funktion : Laufwerk A und B an
- "Verify" : an
- "Fastload" : an

Wer eine andere Voreinstellung haben will, kann dies problemlos an der gekennzeichneten Stelle im Data-Segment des Listings abändern.

Allerdings ist vom Programm aus noch eine wesentlich bessere Möglichkeit vorgesehen, die Voreinstellung zu beeinflussen. Bei der Installation wird im Wurzelverzeichnis des Boot-Laufwerks nach einer Datei mit dem Namen "STC_*.*" gesucht. Falls vorhanden, wird die Extension dieses Dateinamens als Cache-Größe ausgewertet. Mit den noch möglichen vier Buchstaben hinter "STC_" können folgende Einstellungen vorgenommen werden:

- "A" => Cache für Laufwerk A
- "B" => Cache für Laufwerk B
- "V" => "Verify"
- "F" => "Fastload"

Ist der jeweilige Buchstabe im Namen enthalten, wird die Funktion eingeschaltet. Zum Beispiel: "STC_AVF.60" oder auch "STC_AVF.060" bewirken folgende Einstellung => Cache Laufwerk A an, Cache Laufwerk B aus, "Verify" an, "Fastload" an, Cache-Größe 60 KByte. Aber keine Angst, diese Einstellungen sind nicht endgültig, sie können nachträglich wieder geändert werden, indem das Accessory aus der Menüleiste heraus aufgerufen wird. Es erscheint dann eine Box, in der die entsprechenden Änderungen vorgenommen werden können.

Zusätzlich zu den eben genannten Möglichkeiten besteht aus dieser Box heraus aber noch die Möglichkeit, die Cache-Funktion für Laufwerk A und B individuell zu sperren, was bedeutet, daß der Cache-Speicher für das jeweilige Laufwerk praktisch eingefroren wird. Es werden dann zwar noch die im Cache vorhandenen Sektoren ausgelesen, aber keine weiteren mehr in den Cache aufgenommen.

Auch kann die Box über den "LÖSCHEN"-Button verlassen werden, was eine komplette Löschung des Cache-Speichers mit sich führt.

Mit einem kleinen Trick kann diese Löschung aber auch für ein einzelnes Laufwerk erfolgen:

Man schaltet die Cache-Funktion für das gewünschte Laufwerk ab und verläßt die Box über "OK" (jetzt werden die Cache-Daten dieses Laufwerks gelöscht). Danach ruft man das Accessory ein weiteres Mal auf und schaltet die Cache-Funktion wieder ein. Für diejenigen, die mit den Begriffen "Fastload" und "Verify" nichts anfangen können, hier noch eine kleine Erklärung:

Verify

Im RAM ist ab der Adresse $444 die Systemvariable '_fverify' abgelegt. Enthält diese Variable einen anderen Wert als Null, wird nach jedem Floppy-Schreibzugriff ein "Verify" durchgeführt, d.h. ein Sektor wird direkt nach dem Schreibvorgang noch einmal gelesen, um möglicherweise aufgetretene Fehler festzustellen.

Setzt man jetzt diese Systemvariable auf Null, unterläßt das Betriebssystem dieses "Verify", woraus eine Halbierung des Zeitbedarfs für jeden Schreibvorgang folgt. Doch dieser schöne Geschwindigkeitsgewinn führt einen kleinen, aber feinen Nachteil mit sich, denn ein fehlerhaft

beschriebener Sektor wird jetzt nicht mehr sofort bemerkt, worunter natürlich die Datensicherheit leidet. Man sollte das "Verify" also nur abschalten, wenn man einwandfreie Disketten und Laufwerke besitzt.

Fastload

Beim "Fastload" ist die Sache nicht ganz so einfach. Hier gibt es leider keine Systemvariable, die man einfach abändern kann, um die gewünschte Wirkung zu erzielen. Das Betriebssystem besteht bei jedem Track-Wechsel auf einem Vergleich der gewünschten Track-Nummer mit der tatsächlichen Nummer des Tracks, auf dem sich der Schreib/Lese-Kopf befindet ("Seek and Verify").

Bringt man nun das Betriebssystem dazu, daß es auf diesen Vergleich verzichtet, verringert sich die Zeit, die normal für einen Track-Wechsel benötigt wird erheblich (deswegen auch "Fastload"). Der dafür verantwortliche Befehl an den Floppy-Controller steht beim ROM-TOS natürlich auch im ROM und kann somit nicht einfach verändert werden. Wie man diesen Befehl trotzdem abändern kann, erkläre ich später. Auch hier ist die Datensicherheit bei aktiviertem "Fastload" nicht mehr so groß, manche Laufwerke kommen unter Umständen auch gar nicht damit zurecht, hier ist erst einmal Ausprobieren gefragt. Zur Not kann das "Fastload" ja abgeschaltet werden.

Die Arbeitsweise

Das eigentliche Konzept des Programms besteht darin, die Floppy-Schreib-/Lese-Routinen des Betriebssystems abzufangen und eigene Routinen einzuschleifen, die die Cache-Funktionen beherrschen.

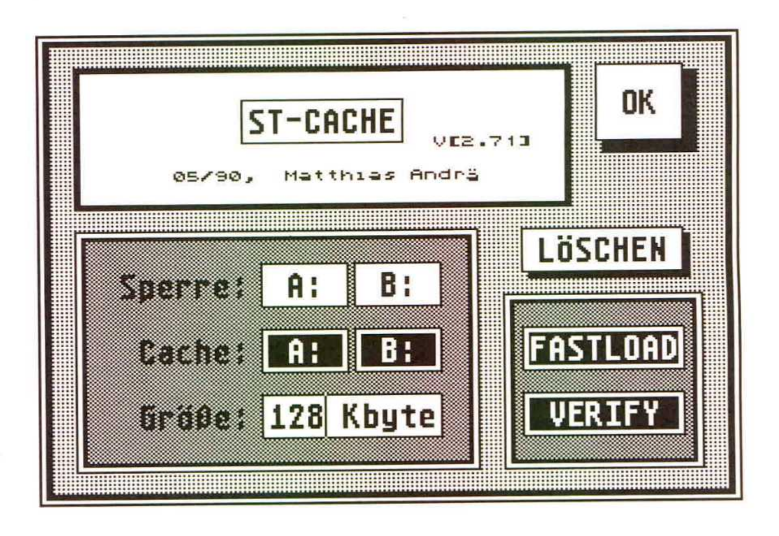
Folgende BIOS- und XBIOS-Routinen sind davon betroffen:

- 'RWABS' (BIOS #4), Sektor(en) lesen/schreiben
- 'MEDIACH' (BIOS #9), Test auf Diskettenwechsel
- 'FLOPRD' (XBIOS #8), Sektor(en) lesen
- 'FLOPWR' (XBIOS #9), Sektor(en) schreiben

Damit die eigenen Routinen später vom Betriebssystem auch angesprungen werden, müssen einige Vektoren "umgebogen" werden.

Die 'RWABS'- und die 'MEDIACH'-Routine kann man man ganz einfach durch Eintragen der neuen Adressen in die Systemvariablen 'hdv_rw' und 'hdv_mediach' umlenken.

Bei den beiden XBIOS-Routinen wird die Sache schon etwas komplizierter, hier müssen der gesamte XBIOS-Trap-Handler mitsamt der Sprungtabelle ins RAM kopiert, der TRAP #14-Vektor auf die neue Adresse im RAM gesetzt und die zwei Adressen von 'FLOPRD' und 'FLOPWR' in der Sprungtabelle des neuen Trap-Handlers ausgetauscht werden. Damit ist es aber leider immer noch nicht getan. Hinter dem XBIOS-Trap-Handler befinden sich die Floppy-Routinen des Betriebssystems, die auch noch komplett ins RAM kopiert werden müssen, um eine Änderung des "Seek and Verify"-Befehls zu gestatten (s.o. "Fastload"). An der entsprechenden Adresse wird dann dieser Befehl in einen "Seek"-Befehl (ohne "Verify") abgeändert. Diese kleine Manipulation halbiert die Zeit, die für einen Track-Wechsel benötigt wird, was ja nicht unbedingt zu verachten ist.



Zum Programm

Die Initialisierung der neuen Floppy-Cache-Routinen nimmt das Unterprogramm 'init' vor. Zuerst werden die Patch-Adressen der jeweiligen TOS-Version in einer Tabelle gesucht. Sind sie gefunden, wird Speicher reserviert, in den die Floppy-Routinen des Betriebssystems anschließend kopiert werden (außer bei Disketten-TOS). Danach werden einige Vektoren umgelegt (s.o.). Mit der GEMDOS-Funktion 'SFIRST' wird dann das Wurzelverzeichnis des Directories nach einer Steuerdatei durchsucht. Ist ein geeigneter File-Name gefunden, wird er ausgewertet und entsprechend Speicher für den Cache reserviert. Ist nichts gefunden worden, wird die Voreinstellung übernommen.

Die Cache-Funktion

Die für das Lesen und Schreiben verantwortlichen Routinen sind 'floprd' und 'flopwr'. Diese beiden Unterprogramme springen eigentlich nur in die Original-Betriebssystemroutinen weiter, wobei aber die Cache-Routinen 'cache_in' und 'cache_out' eingeschleift werden.

In 'cache_in' wird ein soeben von Diskette gelesener Sektor in den Cache-Speicher aufgenommen, vorausgesetzt die Statistik "befürwortet" eine Aufnahme. Hier wird auch ein auf Diskette geschriebener Sektor im Cache wieder gelöscht, da es wahrscheinlich ist, daß er sich durch die Schreiboperation geändert hat. Auf eine mögliche direkte Aufnahme neu geschriebener Sektoren in den Cache habe ich verzichtet, da dies bei dem verwendeten Algorithmus keine sonderlichen Vorteile gebracht hat. In 'cache_out' werden Sektoren, die von Diskette gelesen werden sollen, im Cache gesucht und, falls vorhanden, direkt aus dem Cache geladen. Wenn ein Sektor nicht im Cache ist, wird er natürlich ganz normal von Diskette gelesen.

Das funktioniert jetzt alles solange ganz prima, bis man auf die ja nicht ganz abwegige Idee kommt, seinem Rechner zur Abwechslung einmal eine andere Diskette zu gönnen. Wenn jetzt immer noch brav die Sektoren der vorhergehenden Diskette aus dem Cache geholt werden, ist mit nahezu tödlicher Sicherheit ein "bombiges" Spektakel zu erwarten. Da das wohl nicht unbedingt erstrebenswert ist, müssen bei einem Diskettenwechsel die Statistik und die Cache-Sektoren des betroffenen Laufwerks gelöscht werden. Diese Arbeit verrichtet das Unterprogramm 'disc_changed'. Es ist in die 'rwabs'- und die 'mediach'-Routine eingeschleift und wird aufgerufen, wenn ein Diskettenwechsel erkannt wurde.

Zur Anwendung

Das Programm entfaltet seine volle Geschwindigkeit grundsätzlich immer dann, wenn z.B. zu ladende Programme oder Dateien komplett im Cache enthalten sind. Ist dies der Fall, erreicht ST-Cache beim Laden die Geschwindigkeit einer guten RAM-Disk.

Ein sinnvoller Einsatz ergibt sich z.B. bei der Arbeit mit Programmiersprachen wie Pascal oder C, die einen ständigen Wechsel zwischen Editor, Compiler, Linker o.ä. verlangen.

Auch nicht im Cache enthaltene Programme können etwas schneller geladen werden, da die Directories und FATs der jeweiligen Disketten zumeist gleich als erstes in den Cache aufgenommen werden und dadurch der Zeitaufwand für das Lesen des Directorys gleich Null ist. Die günstigste Größe für den Cache-Speicher muß jeder selbst für sich herausfinden, sie sollte aber so gewählt werden, daß zumindest ein Teil der hauptsächlich benutzten Programme oder Dateien vollständig hineinpaßt (siehe Benchmarks). Aus den Benchmark-Werten kann man entnehmen, daß im günstigsten Fall (alle zu ladenden Sektoren sind bereits im Cache enthalten) eine fast 40fache Geschwindigkeitssteigerung zu erreichen ist.

Fehlermeldungen

Einige während der Installation möglicherweise auftretende Fehler werden unter Angabe einer Nummer vom Programm gemeldet. Diese stehen für folgende Meldungen:

#1 - keine Floppy angeschlossen
#2 - Programm läuft nicht mit der benutzten TOS-Version
#3 - zuwenig freies RAM vorhanden
#4 - Fehler beim Laden des RSC-Files aufgetreten
#5 - Fehler im RSC-File

Nun noch ein paar Einschränkungen:

• Das Accessory läuft in jeder Auflösung, aber nur mit folgenden TOS-Versionen:

- TOS 1.0 vom 06.02.1986 (auch Disketten-TOS)
- TOS 1.2 vom 22.04.1987 (Blitter-TOS)
- TOS 1.4 vom 06.04.1989 (Rainbow-TOS)
- TOS 1.6 vom 19.06.1989 (STE-TOS)
- TOS 1.6 vom 29.07.1989 (STE-TOS)

Bei anderen TOS-Versionen wird eine Fehlermeldung ausgegeben!

• Im Farbmodus sollte die Bildschirmauflösung auf keinen Fall über das Desktop gewechselt werden, da das Accessory dann mehrfach installiert wird.

• Das Programm ist für Diskettenformate bis maximal 87 Tracks und 14 Sektoren pro Track ausgelegt (das dürfte wohl für die meisten halbwegs normalen Diskettenformate ausreichen).

• Bei Benutzung schreibgeschützter Disketten wird vom Betriebssystem vor jeder Diskettenoperation der Boot-Sektor gelesen, um einen möglichen Diskettenwechsel festzustellen. Der Geschwindigkeitsgewinn fällt deswegen mit ST-Cache hier geringer aus.

• Im Desktop verursacht das Öffnen eines Laufwerksfensters oder der Wechsel in einen oder aus einem Ordner die Löschung der Cache-Daten des jeweiligen Laufwerks, da hierbei vom Betriebssystem ein Diskettenwechsel signalisiert wird. (Dies entfällt zumindest bei TOS-Version 1.6.)

• Da sich das Accessory den für den Cache benötigtenb Speicherplatz direkt unterhalb des oberen Speicherendes (Bildschirmspeicher) reserviert, kann es zu Konflikten mit anderen residenten Programmen oder Accessories kommen, die dieses Verfahren auch praktizieren. Ändert man z.B. bei einer RAM-Disk, die auf diese Weise Speicher reserviert, die Größe, sollte man vorher die Cache-Größe auf 0 KByte setzen, da daraufhin der Cache-Speicher freigegeben wird. Umgekehrt kann es aber auch nötig sein, erst die RAM-Disk auf 0 KByte Größe zu setzen, um die Cache-Größe ändern zu können. Dies hängt dann jeweils von der Reihenfolge der Installation ab. Achtung, Zuwiderhandlungen können mit scheinbar unerklärlichem Speicherschwund geahndet werden!

Zu den Listings

Das Listing des Accessorys wurde mit dem GST-Macro-Assembler erstellt, sollte auf anderen Assembler-Systemen aber auch problemlos übersetzt werden können, wenn man deren jeweilige Eigenheiten beachtet. Es kann also sein, daß die eine oder andere Assembler-Direktive noch zusätzlich verlangt wird (z.B. '.text','.data','.even') oder etwas abgeändert werden muß. Für alle, die sich schon ein klein wenig mit Assembler beschäftigt haben, dürfte das aber wohl kaum ein Problem darstellen. Die Kommentare können natürlich alle weggelassen werden. Das zweite Listing stellt ein GFA-BASIC-Programm dar, das das vom Accessory unbedingt benötigte RSC-File unter dem Namen "ST_CACHE.RSC" erzeugt.

Benchmark 1: *80 Tracks zu 9 Sektoren nacheinander lesen (360 kB)*

TOS "pur" : ca. 32 Sekunden
Mit Cache ("Fastload" an):

Cache-Größe in KByte	1. Lesen Zeit in Sek.	weiteres Lesen Zeit in Sek.
0	16.2	16.2
50	16.2	14.1
100	16.2	11.2
150	16.2	9.8
200	16.2	7.8
250	16.3	5.7
300	18.5	3.6

Benchmark 2: *Datei laden*

1. Dateigröße ca. 56 KByte

TOS "pur" : ca. 5.3 Sekunden
Mit Cache ("Fastload" an):

Cache-Größe in KByte	1. Laden Zeit in Sek.	weiteres Laden Zeit in Sek.
0	2.9	2.9
50	2.9	2.2
60	2.9	0.13

2. Dateigröße ca. 130 KByte

TOS "pur" : ca. 12.6 Sekunden
Mit Cache ("Fastload" an):

Cache-größe in KByte	1. Laden Zeit in Sek.	weiteres Laden Zeit in Sek.
0	6.6	6.6
50	6.6	6.0
100	6.6	3.6
130	6.6	0.35

Und noch ein Tip zum Schluß

Wer das Programm zum Laufen gebracht hat, sollte es nicht gleich mit seinen "teuersten" Disketten ausprobieren, geschweige denn mit Disketten von denen kein Backup existiert, da ein Tippfehler sich evtl. fatal auswirken kann. Also am besten ein oder zwei leere Disketten genommen, diese randvoll mit irgendwelchen Daten gepackt und das Accessory gründlich ausgetestet (z.B. Dateien kopieren, löschen, Programme starten, Disketten wechseln, Cache-Einstellung verändern usw.). Wenn am Ende immer noch alle Dateien so aussehen, wie sie es vorher getan haben, und der Rechner nicht dauernd abgestürzt ist, kann es mit den Fehlern, die sich vielleicht beim Abtippen eingeschlichen haben, nicht so schlimm sein.

Matthias Andrä

```
  1: **************************************************
  2: *                    ST-CACHE                    *
  3: *                                                *
  4: *           Version 2.71 ,    05/90              *
  5: *                                                *
  6: *       Für:  - TOS 1.0 vom 06.02.1986           *
  7: *             - TOS 1.2 vom 22.04.1987           *
  8: *             - TOS 1.4 vom 06.04.1989           *
  9: *             - TOS 1.6 vom 19.06.1989           *
 10: *             - TOS 1.6 vom 29.07.1989           *
 11: *                                                *
 12: *       Sprache:  GST Macro-Assembler            *
 13: *                                                *
 14: *              Matthias Andrä                    *
 15: *                                                *
 16: *          (c) MAXON Computer GmbH               *
 17: *                                                *
 18: **************************************************
 19: ;
 20:  SECTION    st_cache
 21: ;
 22: start:
 23:  lea        stack,sp      ; Stackadresse setzen
 24:  bsr        appl_init     ; Anwendung anmelden
 25:  bsr        rsrc_load     ; RSC-File laden
 26:  tst.b      fehler        ; Fehler aufgetreten?
 27:  bne.s      do_box
 28:  clr.w      d0
 29:  clr.w      d1
 30:  bsr        rsrc_gaddr
 31:  tst.b      fehler        ; Fehler?
 32:  bne.s      do_box
 33:  move.l     addrout,rsrcaddr ; Adresse RSC-Daten
 34:  bsr        form_center   ; RSC-Box zentrieren
 35:  bsr        init          ; Cache initialisieren
 36:  tst.b      fehler
 37:  bne.s      do_box
 38:  bsr        menu_register ; Accessory eintragen
 39: ;
 40: acc_loop:
 41:  bsr        evnt_mesag
 42:  cmp.w      #40,ev_buff   ; Accessory gewählt?
 43:  bne.s      acc_loop      ; nein, zurück
 44:  move.w     ev_buff+8,d0
 45:  cmp.w      acc_id,d0     ; Cache-Accessory?
 46:  bne.s      acc_loop      ; nein, ein anderes
 47: do_box:
 48:  bsr.s      main          ; ja, RSC-Box bearbeiten
 49:  bra.s      acc_loop      ; und zurück an Anfang
 50: ;
 51: main:
 52:  lea        err_txt,a0    ; Adresse Fehlermeldung
 53:  tst.b      fehler        ; Installationsfehler?
 54:  bne.l      form_alert    ; ja, Meldung und zurück
 55: ;
 56:  bsr        write_size    ; Cache-Grösse schreiben
 57:  clr.w      d0
 58:  bsr        form_dial     ; Bildbereich
                              reservieren
 59:  moveq.l    #1,d0         ; sich vergrösserndes
 60:  bsr        form_dial     ; Rechteck und
 61:  bsr        obj_draw      ; Abfrage-Box zeichnen
 62:  bsr        form_do       ; bearbeiten
 63:  bsr        get_size      ; Cachegrösse lesen
 64:  bsr        reserve       ; Speicher reservieren
 65:  move.l     rsrcaddr,a1   ; Adresse RSC-Daten
 66:  move.w     #$30,466(a1)  ; Ok-Button 'normal'
 67:  move.l     buffer,a0     ; Adresse kopiertes ROM
 68:  add.w      f_load,a0     ; +Offset f. Fastload
 69:  move.b     #$10,(a0)     ; Seek ohne Verify
 70:  cmp.w      #$11,394(a1)  ; Fastload an?
 71:  beq.s      main5         ; ja, weiter
 72:  move.b     #$14,(a0)     ; nein, Seek mit Verify
 73: main5:
 74:  cmp.w      #$31,442(a1)  ; Cachedaten löschen?
 75:  bne.s      main6         ; nein, weiter
 76:  bsr        cache_del     ; ja, löschen
 77: main6:
 78:  move.w     #$30,442(a1)  ; Lösch-Button 'normal'
 79:  move.w     #3,laufwerk   ; Cache beide Drive's
 80:  cmp.w      #$11,154(a1)  ; Cache Drive A an?
 81:  beq.s      main7         ; ja, weiter
 82:  and.w      #2,laufwerk   ; nein, Cache aus
 83:  clr.w      d3            ; und von Drive A
 84:  bsr        disc_changed  ; die Cachedaten löschen
 85: main7:
 86:  cmp.w      #$11,178(a1)  ; Cache Drive B an?
 87:  beq.s      main8         ; ja, weiter
 88:  and.w      #1,laufwerk   ; nein, Cache aus
 89:  moveq.l    #1,d3         ; und von Drive B
 90:  bsr        disc_changed  ; die Cachedaten löschen
 91: main8:
 92:  clr.w      sperre        ; Sperre Drive A und B
                                an
 93:  cmp.w      #$11,106(a1)  ; Sperre Drive A an?
 94:  beq.s      main9         ; ja, weiter
 95:  move.w     #1,sperre     ; nein, ausschalten
 96: main9:
 97:  cmp.w      #$11,202(a1)  ; Sperre Drive B an?
 98:  beq.s      main10        ; ja, weiter
 99:  or.w       #2,sperre     ; nein, ausschalten
100: main10:
101:  moveq.l    #3,d0         ; sich verkleinerndes
102:  bsr        form_dial     ; Rechteck zeichnen
103:  moveq.l    #2,d0         ; Bildbereich freigeben
104:  bra.l      form_dial     ; und zurück
105: ;
106: ; Umwandlung von 'maxsec' für Grössenangabe
107: write_size:
108:  moveq.l    #2,d0
109:  clr.w      d1
110:  bsr        rsrc_gaddr
111:  move.l     addrout,a1
112:  move.l     (a1),a1       ; Adresse Grössenangabe
113:  addq.l     #3,a1
114:  move.w     maxsec,d0     ; Anzahl der Sektoren
115:  lsr.w      #1,d0         ; durch 2 = Grösse in KB
116:  ext.l      d0            ; auf Langwortbreite
117:  moveq.l    #2,d1         ; 3 Stellen
118: ws_loop:
119:  divu       #10,d0        ; Grösse durch 10
120:  swap       d0            ; Rest in d0.w
121:  add.b      #'0',d0       ; + ASCII-CODE von '0'
122:  move.b     d0,-(a1)      ; Ziffer schreiben
123:  clr.w      d0            ; Rest löschen
124:  swap       d0            ; und wieder tauschen
125:  dbra.w     d1,ws_loop    ; nächste Ziffer
126:  rts
127: ;
128: getsysbase:
129:  move.l     $4f2,a0
130:  move.l     a0,sysbase    ; Startadresse TOS
131:  cmp.w      #$102,2(a0)   ; TOS-Version < 1.2?
132:  blt.s      pd_ok         ; ja, weiter
133:  move.l     40(a0),pd     ; Process-Descriptor
134: pd_ok:
135:  tst.w      $4a6          ; mind. eine Floppy?
136:  bne.s      gs_end        ; ja, weiter
137:  move.b     #1+48,fehler  ; nein, Fehler
138: gs_end:
139:  rts                      ; und zurück
140: ;
141: init:
142:  pea        getsysbase    ; getsysbase-Routine
143:  move.w     #38,-(sp)     ; im Supervisormodus
144:  trap       #14           ; ausführen
145:  addq.l     #6,sp
146:  tst.b      fehler        ; Fehler?
147:  beq.s      init_1        ; nein, weiter
148:  rts                      ; ja, zurück
149: init_1:
150:  move.l     sysbase,a4    ; TOS-Startadresse in a4
151:  move.l     a4,d4         ; und d4
152:  move.l     $18(a4),d1    ; TOS Erstellungsdatum
153:  lea        tos_adr(pc),a3 ; Start
                                 Adressentabelle
154:  move.w     (a3)+,d2      ; Anzahl Adressblöcke
155: adr_loop:
156:  cmp.l      (a3)+,d1      ; gleiches Datum?
157:  bne.s      nxt_vers      ; nein, weitersuchen
158:  cmp.l      (a3),d4       ; gleiche 'sysbase'?
159:  beq.s      patch         ; ja, patchen
160: nxt_vers:
161:  lea        20(a3),a3     ; nächster Adressblock
162:  dbra.w     d2,adr_loop
163:  move.b     #2+48,fehler  ; TOS-Version unbekannt
164:  rts                      ; zurück
165: ;
166: patch:
167:  addq.l     #4,a3         ; a3 auf Anfang Patch-
                                Adresse
```

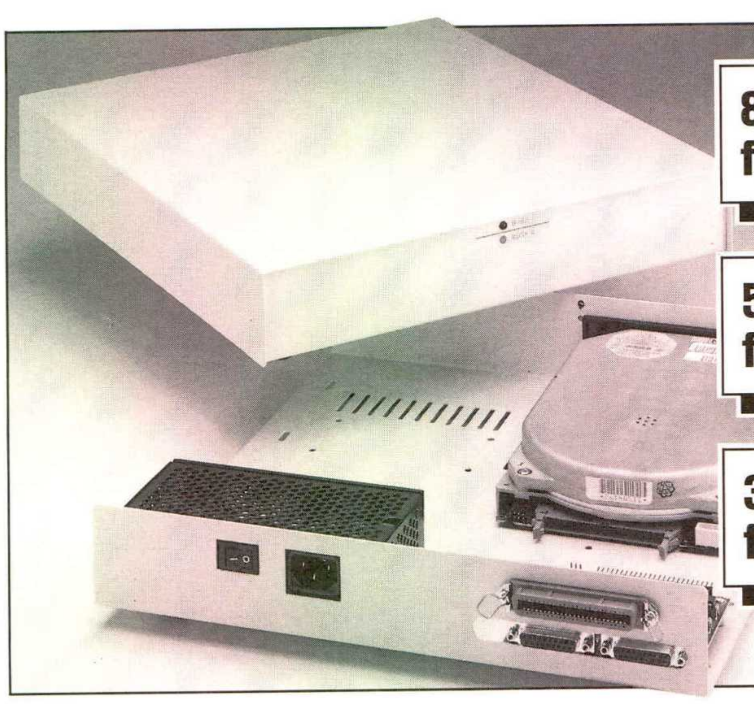
→

```
168:  move.w    (a3)+,trap14 ; Adresse Trap-Handler
169:  move.w    (a3)+,f_load ; Adresse 'Fastload-
                                        Byte'
170:  move.w    (a3)+,rw_abs ; Adresse 'rwabs'-
                                        Routine
171:  cmpa.w    #$6100,a4    ; Disketten-TOS 1.0?
172:  beq.s     patch_d      ; ja, weiter
173:  move.l    #$2070,-(sp) ; nein, ca. 8 KB für
                               Kopie
174:  move.w    #$48,-(sp)
175:  trap      #1           ; anfordern (Malloc)
176:  addq.l    #6,sp
177:  tst.l     d0           ; Ok?
178:  bpl.s     copy_tos     ; ja, weiter
179:  move.b    #3+48,fehler ; nein, Fehler
180:  rts                    ; und zurück
181: patch_d:
182:  move.l    a4,d0        ; Startadresse Disk.-TOS
183: copy_tos:
184:  move.l    d0,buffer    ; Startadresse merken
185:  pea       copy         ; Floppyroutinen aus
186:  move.w    #38,-(sp)    ; TOS kopieren (ROM-TOS)
187:  trap      #14          ; (im Supervisormodus)
188:  addq.l    #6,sp
189:  move.l    buffer,a2    ; Startadresse der Kopie
190:  move.l    a2,d2
191:  move.l    a2,a1        ; (im TOS: 'jsr
                                   flopver')
192:  add.w     (a3)+,a1     ; Adresse relozieren
193:  sub.l     d4,(a1)      ; Adresse relozieren
194:  add.l     d2,(a1)      ; Adresse relozieren
195:  move.l    a2,a1        ; (im TOS: 'jsr floprd')
196:  add.w     (a3)+,a1     ; Adresse relozieren
197:  sub.l     d4,(a1)      ; Adresse relozieren
198:  add.l     d2,(a1)      ; Adresse relozieren
199:  move.l    (a1),flp_rd  ; Adresse floprd merken
200:  move.l    a2,a1        ; (im TOS: 'jsr flopwr')
201:  add.w     (a3)+,a1     ; Adresse relozieren
202:  sub.l     d4,(a1)      ; Adresse relozieren
203:  add.l     d2,(a1)      ; Adresse relozieren
204:  move.l    (a1),flp_wr  ; Adresse flopwr merken
205:  move.l    #flopwr,(a1) ; eigene Routine setzen
206:  move.l    a2,a1        ; (im TOS:
                                  'jsr floprd')
207:  add.w     (a3)+,a1     ; Adresse relozieren
208:  sub.l     d4,(a1)      ; Adresse relozieren
209:  add.l     d2,(a1)      ; Adresse relozieren
210:  move.l    #floprd,(a1) ; eigene Routine setzen
211:  move.l    a2,a1
212:  add.w     (a3),a1
213:  move.l    #floprd,(a1)+ ; Trap #14 FLOPRD- und
214:  move.l    #flopwr,(a1) ; FLOPWR-Routinen
                               umlegen
215:  ; Auswertung der Cache-Steuerdatei
216:  pea       dta
217:  move.w    #$1a,-(sp)
218:  trap      #1           ; DTA setzen
219:  addq.l    #6,sp
220:  clr.w     -(sp)
221:  pea       f_name       ; Steuerdatei-Name lesen
222:  move.w    #$4e,-(sp)
223:  trap      #1
224:  addq.l    #8,sp
225:  lea       dta+30,a0   ; Adresse Steuerdatei-
                                   Name
226:  tst.w     d0          ; Name auf Disk gefunden?
227:  beq.s     sf_found    ; ja, weiter
228:  lea       filename,a0 ; nein, Voreinstellung
229:  sf_found:
230:  moveq.l   #8,d7        ; in max. 9 Stellen
231:  fn_loop:
232:  cmp.b     #'.',(a0)+   ; '.' suchen
233:  dbeq.w    d7,fn_loop
234:  ; Grössenangabe aus der Steuerdatei einlesen
235:  move.l    a0,a1
236:  lea       -1(a0),a2
237:  bsr       wandel
238:  ; und weitere Informationen daraus verarbeiten
239:  clr.w     laufwerk
240:  move.l    rsrcaddr,a1
241:  moveq.l   #3,d7
242:  cdefloop:
243:  move.b    -(a2),d0
244:  cmp.b     #'V',d0      ; 'V' in Dateiname?
245:  bne.s     cdef1
246:  move.w    #$11,418(a1) ; ja, Verify an
247:  cdef1:
248:   cmp.b    #'A',d0      ; 'A' in Dateiname?
249:   bne.s    cdef2
250:   move.w   #$11,154(a1) ; ja, Cache an, Drive A
251:   or.w     #1,laufwerk
252:  cdef2:
253:   cmp.b    #'B',d0      ; 'B' in Dateiname?
254:   bne.s    cdef3
255:   move.w   #$11,178(a1) ; ja, Cache an, Drive B
256:   or.w     #2,laufwerk
257:  cdef3:
258:   cmp.b    #'F',d0      ; 'F' in Dateiname?
259:   bne.s    cdef4
260:   move.w   #$11,394(a1) ; ja, Fastload an
261:   move.l   buffer,a0    ; in Adresse des
262:   add.w    f_load,a0    ; "Fastload"-Bytes
263:   move.b   #$10,(a0)    ; Seek ohne Verify
264:  cdef4:                 ; setzen
265:   cmp.b    #'_',d0      ; Abbruch bei '_'
266:  cdefnxt:
267:   dbeq.w   d7,cdefloop  ; nächstes Zeichen
268:  ;
269:  reserve:
270:   move.w   oldmxsec,d0  ; Cache-Grösse
271:   cmp.w    maxsec,d0    ; geändert?
272:   bne.s    res1         ; ja, weiter
273:   rts                   ; nein, Abbruch
274:  res1:
275:   move.w   #-1,oldmxsec
276:   tst.l    stat         ; schon RAM reserviert?
277:   beq.s    no_mfree     ; nein, weiter
278:   bsr      mfree        ; ja, altes RAM
                               freigeben
279:   clr.l    stat         ; alte Adresse löschen
280:  no_mfree:
281:   tst.w    maxsec       ; Cachegrösse <> 0 KB
282:   bne.s    reservl      ; Ja, weiter
283:  no_cache:
284:   move.l   rsrcaddr,a0
285:   move.w   #$10,154(a0) ; Cache Drive A
286:   move.w   #$10,178(a0) ; und Drive B aus
287:   clr.w    laufwerk
288:   rts
289:  reservl:
290:   move.l   #-1,-(sp)    ; wieviel RAM ist frei?
291:   move.w   #$48,-(sp)
292:   trap     #1           ; Malloc (als Abfrage)
293:   addq.l   #6,sp
294:   sub.l    #10000,d0    ; 10000 Bytes min. Rest?
295:   bmi.s    res_err      ; nicht genug, Fehler
296:   divu     #516,d0
297:   cmp.w    #2,d0        ; mindestens 2 Sektoren?
298:   blt.s    res_err      ; nein, zur
                               Fehlermeldung
299:   cmp.w    maxsec,d0    ; Grösse möglich?
300:   bge.s    reserv2      ; ja, weiter
301:   move.w   d0,maxsec    ; max. Grösse nehmen
302:  reserv2:
303:   move.w   maxsec,d0    ; Cachegrösse
304:   mulu     #516,d0      ; * Grösse Cachesektor
305:   add.l    #4872+255,d0 ; + Statistikbedarf
306:   clr.b    d0           ; auf 256 Byte-Grenze
307:   bsr.s    malloc       ; Speicher reservieren
308:   move.l   d0,stat      ; Startadresse merken
309:   add.l    #4872,d0
310:   move.l   d0,cache     ; Anfang Cachesektoren
311:   move.w   maxsec,oldmxsec
312:   bra.l    cache_del    ; Cache initialisieren
313:  ;                         und zurück
314:  res_err:
315:   clr.w    maxsec       ; keine Cachesektoren
316:   bra.s    no_cache     ; Cache Drive A+B aus
317:  ;
318:  ; spezielles Malloc für Accessories
319:  malloc:
320:   movem.l  d1-d5/a0-a3/a5,-(sp)
321:   move.l   d0,d4
322:   move.l   pd,a5      ; Adresse Process-
                               Descriptor
323:   move.l   (a5),d5  ; Process-Descriptor (PD)
324:   move.l   #start-256,(a5) ; eigenen einsetzen
325:   move.l   #-1,-(sp)
326:   move.w   #$48,-(sp)   ; Wieviel RAM ist frei?
327:   trap     #1
328:   sub.l    d4,d0        ; - gewünschte Länge
329:   move.l   d0,2(sp)     ; wieder auf Stack    →
```

```
330:   trap      #1            ; Speicher reservieren
331:   move.l    d0,a3         ; Startadresse merken
332:   move.l    d4,2(sp)      ; restl. Speicherblock
333:   trap      #1            ; am Speicherende
                                 reserviert
334:   addq.l    #6,sp
335:   move.l    d0,d4         ; Startadresse merken
336:   move.l    a3,-(sp)      ; ersten Block
337:   move.w    #$49,-(sp)
338:   trap      #1            ; wieder freigeben
339:   addq.l    #6,sp
340:   move.l    d5,(a5)       ; alten PD einsetzen
341:   move.l    d4,d0         ; Startadresse nach d0
342:   movem.l   (sp)+,d1-d5/a0-a3/a5
343:   rts
344:  ; spezielles Mfree
345:  mfree:
346:   movem.l   d1-d2/d5/a0-a3/a5,-(sp)
347:   move.l    pd,a5
348:   move.l    (a5),d5       ; akt. PD merken
349:   move.l    #start-256,(a5) ; eigenen PD
                                einsetzen
350:   move.l    stat,-(sp)    ; Cachespeicher
351:   move.w    #$49,-(sp)
352:   trap      #1            ; freigeben
353:   addq.l    #6,sp
354:   move.l    d5,(a5)       ; alten PD zurückholen
355:   movem.l   (sp)+,d1-d2/d5/a0-a3/a5
356:   rts
357:  ;
358:  copy:
359:   move.l    a4,a0         ; Startadresse TOS
360:   move.l    buffer,a1     ; Zieladresse im RAM
361:   cmp.l     a0,a1         ; Disketten-TOS?
362:   beq.s     eintrag       ; ja, weiter
363:   move.w    #2075,d7      ; nein, ca. 8 KBytes TOS
364:  bb_loop:
365:   move.l    (a0)+,(a1)+   ; kopieren
366:   dbra.w    d7,bb_loop
367:  eintrag:
368:   move.w    #$476,a0      ; 'rwabs'-Vektor
369:   move.l    (a0),old_rwabs ; merken
370:   move.l    #rwabs,(a0)   ; eigene Routine setzen
371:   move.l    8(a0),old_mediach ; 'mediach'-Vektor
372:   move.l    #mediach,8(a0) ; eigene Routine
                               setzen
373:   move.l    buffer,a1
374:   add.w     trap14,a1
375:   move.l    a1,$b8        ; Trap #14 umlegen
376:   rts
377:  ;
378:  ; Grössenangabe aus RSC-File lesen
379:  get_size:
380:   moveq.l   #2,d0
381:   clr.w     d1
382:   bsr       rsrc_gaddr
383:   move.l    addrout,a1
384:   move.l    (a1),a1       ; Adresse Grössenangabe
385:  wandel:
386:   moveq.l   #2,d1         ; 3 Stellen (max. 999)
387:   clr.w     d2
388:   moveq.l   #0,d0         ; Zahl löschen
389:  w_loop:
390:   move.b    (a1)+,d2      ; Zeichen holen
391:   cmp.b     #48,d2        ; Zeichen < '0'?
392:   blt.s     w_end
393:   cmp.b     #57,d2        ; > '9'?
394:   bgt.s     w_end
395:   sub.b     #48,d2        ; - ASC('0')
396:   add.w     d2,d0         ; Einerstelle addieren
397:   mulu      #10,d0        ; *10
398:  w_end:
399:   dbra.w    d1,w_loop     ; nächste Stelle
400:   divu      #10,d0        ; durch 10 = Grösse in
                                              KB
401:   add.w     d0,d0         ; *2
402:   move.w    d0,maxsec     ; = Anzahl Cachesektoren
403:   rts
404:  ;
405:  ; Einsprung aus 'floprd'
406:  cache_out:
407:   lea       regsave,a0
408:   movem.l   d1-d3/a1-a3,-(a0)
409:   move.w    drive,d2      ; Laufwerk
410:   addq.w    #1,d2
411:   and.w     laufwerk,d2   ; Cache von Drive aus?
412:   beq.s     cr_end        ; ja, Abbruch
413:   bsr       get_para      ; Parameter holen
414:   move.l    stat,a3       ; Statistikadresse
415:   add.w     d1,a3         ; Offset addieren
416:  c_outloop:
417:   move.l    cache,a1      ; Adresse Cachesektoren
418:   lea       -514(a1),a1   ; zurücksetzen
419:   move.w    maxsec,d2     ; Anzahl Sektoren
420:   subq.w    #1,d2         ; minus 1
421:  cout1:
422:   lea       516(a1),a1
423:   cmp.w     (a1),d1       ; Sektor in Cache?
424:   dbeq.w    d2,cout1
425:   bne.s     cr_end        ; nein, dann Ende
426:  co_found:
427:   addq.l    #2,a1
428:   addq.b    #1,-4(a1)     ; Statistik +1
429:   move.l    8(sp),a2      ; Bufferadresse
430:   move.l    a2,d3
431:   bsr.s     fastcopy
432:  nxt_sec:
433:   addq.b    #1,(a3)       ; Statistik +1
434:   bsr       stat_reset    ; Statistiküberlauf?
435:   addq.l    #4,a3         ; nächste
                                Statistikadresse
436:   addq.w    #4,d1         ; nächste relative
                                Statistikadresse
437:   add.l     #512,8(sp)    ; Bufferadresse+512
438:   addq.w    #1,18(sp)     ; Sektor-Nr. plus 1
439:   subq.w    #1,24(sp)     ; Anzahl Sektoren minus 1
440:   bne.s     c_outloop     ; war's das ?
441:   moveq.l   #0,d0         ; Ok
442:  cr_end:
443:   movem.l   (a0)+,d1-d3/a1-a3
444:   rts
445:  ;
446:  fastcopy:
447:   btst      #0,d3         ; Ist a2 (d3) gerade?
448:   bne.s     bytecopy      ; nein, dann Bytecopy
449:   moveq.l   #15,d3
450:  longcopy:
451:   move.l    (a1)+,(a2)+   ; 512 Bytes aus Buffer
452:   move.l    (a1)+,(a2)+   ; in Cache schreiben
453:   move.l    (a1)+,(a2)+
454:   move.l    (a1)+,(a2)+
455:   move.l    (a1)+,(a2)+
456:   move.l    (a1)+,(a2)+
457:   move.l    (a1)+,(a2)+
458:   move.l    (a1)+,(a2)+
459:   dbra.w    d3,longcopy
460:   rts
461:  ; Byteweises Kopieren (für ungerade Adressen)
462:  bytecopy:
463:   moveq.l   #63,d3
464:  f_bcloop:
465:   move.b    (a1)+,(a2)+
466:   move.b    (a1)+,(a2)+
467:   move.b    (a1)+,(a2)+
468:   move.b    (a1)+,(a2)+
469:   move.b    (a1)+,(a2)+
470:   move.b    (a1)+,(a2)+
471:   move.b    (a1)+,(a2)+
472:   move.b    (a1)+,(a2)+
473:   dbra.w    d3,f_bcloop
474:   rts
475:  ;
476:  get_para:               ; Parameter vom Stack
                                holen
477:   subq.w    #1,d2
478:   add.w     d2,d2         ; Laufwerk*2
479:   add.w     26(sp),d2     ; +Seite
480:   move.w    22(sp),d1     ; Sektor
481:   subq.w    #1,d1
482:   lsl.w     #2,d1         ; *4
483:   add.w     d2,d1         ; +Laufwerk+Seite
484:   move.w    24(sp),d2     ; Track
485:   mulu      #14*4,d2      ; *14 Sektoren
486:   add.w     d2,d1         ; +Sektoradresse
487:   rts
488:  ;
489:  ; Einsprung aus 'floprd'
490:  cache_in:
491:   sf        write         ; Leseoperation
492:   bra.s     cinstart
493:  ; Einsprung aus 'flopwr'
494:  cache2in:
```

→

SCSI-Festplatten zu »Schotten-Preisen«!



Zum Beispiel:

85 MB SCSI-Festplatte (28 ms) für nur DM 1.398,–

50 MB SCSI-Festplatte (40 ms) für nur DM 1.198,–

30 MB SCSI-Festplatte (40 ms) für nur DM 998,–

Unsere SCSI-Festplatten werden komplett anschlußfertig incl. Software und Kabel ausgeliefert.

Ausstattung und Leistungsmerkmale unserer Festplatten:

- Datentransferraten >600 KByte/s (mit CDC- und Maxtorlaufwerken bis zu 850 KByte/s erzielbar), mittlere Zugriffszeiten bis zu 14 ms
- Spitzensoftware: 255 Partitionen installierbar, Passwortfunktion, jede Partition autobootfähig, Interleave 1:1 einstellbar, Cache, Backup, Optimizer in der Software enthalten
- 100% Atari-kompatibel, sämtliche Fremdbetriebssysteme (PC-Speed, PC-Ditto, Spectre, Aladin, Minix, OS-9, RTOS) sind voll lauffähig
- Superleise (3,5″-Festplatten ohne Lüfter, 5,25″-Festplatten mit thermogeregeltem Lüfter)
- Durchgeschleifter gepufferter DMA-Bus, Autoparkfunktion hardwaremäßig
- Herausgeführter SCSI-Bus (50-poliger Centronics-Anschluß, Apple MacIntosh und PC's anschließbar)
- Zweite SCSI-Festplatte im Gehäuse nachrüstbar (SCSI-Hostadapter und Gehäuse für interne zweite Festplatte vorbereitet)
- Unsere SCSI-Festplatten werden komplett anschlußfertig im Gehäuse incl. Netz-, DMA-Kabel, Software und Handbuch geliefert

Preise:

32 MB, 40 ms, ST138N-0	DM 998,–
40 MB, 19 ms, Quantum	DM 1.298,–
49 MB, 40 ms, ST157N-0	DM 1.198,–
85 MB, 28 ms, ST296N	DM 1.398,–
80 MB, 24 ms, ST1096N	DM 1.498,–
80 MB, 19 ms, Quantum	DM 1.798,–
170 MB, 28 ms, 2x ST296N	DM 2.498,–
280 MB, 17 ms, Maxtor	DM 3.498,–
380 MB, 17 ms, Maxtor	DM 3.998,–
702 MB, 14 ms, CDC	DM 5.998,–
1200 MB, 14 ms, CDC	DM 11.998,–
44 MB, 25 ms, SQ 555	DM 1.898,–
SCSI-Kits (Festplatte und SCSI-Hostadapter für ST):	
32 MB Kit (ST138N-0)	DM 848,–
40 MB Kit (P40S)	DM 1.098,–
49 MB Kit (ST157N-0)	DM 998,–
85 MB Kit (ST296N)	DM 1.198,–
80 MB Kit (ST1096N)	DM 1.298,–
80 MB Kit (P80S)	DM 1.598,–
SCSI-Hostadapter (incl. Software und DMA-Kabel)	DM 198,–
DMA-Kabel	DM 39,–
SCSI-Kabel	DM 39,–
Netzteil 50 W	DM 99,–
Gehäuse	DM 99,–
Cartridge für SQ555	DM 239,–

Weitere Modelle sowie sonstige Soft- und Hardware auf Anfrage!



Datensysteme

Eugenstraße 28
7302 Ostfildern 4

Telefon 0711/4579623
Telefax 0711/4569566

```
495:  st        write           ; Schreiboperation
496: cinstart:
497:  lea       regsave,a0
498:  movem.l   d0-d7/a1-a4/a6,-(a0)
499:  move.w    drive,d2        ; Laufwerk
500:  addq.w    #1,d2
501:  and.w     laufwerk,d2     ; Cache von Drive aus?
502:  beq.l     sc_end          ; ja, Abbruch
503:  tst.b     write           ; Schreiboperation?
504:  bne.s     clin            ; ja, Sperre ignorieren
505:  and.w     sperre,d2       ; Cache gesperrt?
506:  beq.l     sc_end          ; ja, Abbruch
507: clin:
508:  bsr.s     get_para        ; Parameter holen
509:  move.l    stat,a4         ; Statistikadresse
510:  add.w     d1,a4           ; Offset addieren
511:  move.w    24(sp),d5       ; Anzahl der Sektoren
512:  subq.w    #1,d5           ; -1 (für Schleife)
513:  move.l    8(sp),a3        ; Bufferadresse
514: c_inloop:
515:  addq.b    #1,(a4)         ; Statistik+1
516:  bsr       stat_reset      ; Statistiküberlauf?
517: ;
518:  tst.b     write           ; Leseoperation?
519:  beq.s     c_read          ; dann, weiter
520: ; *** nur bei Schreiboperationen ***
521:  move.l    cache,a2        ; Adresse Cachesektoren
522:  lea       -514(a2),a2     ; zurücksetzen
523:  move.w    maxsec,d2       ; Anzahl Sektoren
524:  subq.w    #1,d2           ; minus 1
525: cin1:
526:  lea       516(a2),a2
527:  cmp.w     (a2),d1         ; Sektor im Cache?
528:  dbeq.w    d2,cin1         ; nein, nächsten testen
529:  bne.s     nxt_ci          ; nichts, nächster
                                  Sektor
530:  move.l    #$FFFF,-2(a2) ; Cachesektor löschen
531:  bra.s     nxt_ci          ; nächster Sektor
532: ; *** nur bei Leseoperationen ***
533: c_read:
534:  move.b    (a4),d4         ; Statistikwert
535:  cmp.b     min_stat,d4     ; > Statistik-Minimum?
536:  ble.s     nxt_ci          ; nein, nächster Sektor
537:  move.b    d4,d6
538:  move.w    maxsec,d7       ; ges. Anzahl Sektoren
539:  subq.w    #1,d7
540:  move.l    cache,a2        ; Anfang Cachesektoren
541:  lea       -516(a2),a2     ; eine Sektorlänge
                                zurück
542: secloop:
543:  lea       516(a2),a2      ; nächster Sektor
544:  cmp.b     (a2),d6         ; Sektorstatistikwert <=
                                Cachestatistikwert?
545:  dbgt.w    d7,secloop
546:  tst.w     d7              ; alle Sektoren getestet?
547:  bmi.s     scl_end         ; ja, dann weiter
548:  move.b    (a2),d6         ; Statistikwert
                                übernehmen
549:  bne.s     sc_belegt       ; Sektor belegt?
550:  clr.b     min_stat        ; Statistik-Minimum auf 0
551:  bra.s     sc_copy         ; und ab zum Kopieren
552: sc_belegt:
553:  move.l    a2,a6
554:  subq.w    #1,d7           ; Sektorzähler -1
555:  bpl.s     secloop         ; schon alle Sektoren?
556: scl_end:
557:  move.b    d6,min_stat ; Statistik-Minimum
558:  cmp.b     d4,d6           ; Sektor gefunden?
559:  bge.s     nxt_ci          ; nein
560:  move.l    a6,a2           ; ja, Sektoradresse
                                  merken
561: sc_copy:
562:  move.b    (a4),(a2)       ; Statistikwert vor Cache-
                                Sektor schreiben
563:  move.w    d1,2(a2)        ; und auch die relative
                                Statistikadressse
564:  move.l    a3,a1           ; Bufferadresse
565:  move.l    a1,d3           ; nach d3
566:  addq.l    #4,a2           ; Start der Cachedaten
567:  bsr       fastcopy        ; Buffer in Cache
                                kopieren
568: nxt_ci:
569:  addq.l    #4,a4           ; nächste
                                Statistikadresse
570:  addq.w    #4,d1           ; nächste relative
                                Statistikadresse
571:  lea       512(a3),a3    ; Bufferadresse +512
572:  dbra.w    d5,c_inloop ; nächster Sektor
573: sc_end:
574:  movem.l   (a0)+,d0-d7/a1-a4/a6
575:  rts
576: ;
577: ; Daten einer gewechselten Diskette löschen
578: disc_changed:
579:  tst.w     maxsec          ; Cachegrösse=0?
580:  beq.s     dc_end          ; dann Abbruch
581:  move.l    stat,a0         ; Statistikadresse
582:  add.w     d3,d3           ; Drive-Nr. *2
583:  add.w     d3,a0           ; Basisadresse von Drive
584: dc_anf:
585:  clr.b     min_stat        ; Statistikminimum auf 0
586:  move.w    #1217,d2        ; 1218 *2
                                Statistiksektoren
587: dc_del:
588:  clr.w     (a0)            ; Statistikwerte löschen
589:  addq.w    #4,a0           ; nächster Sektor
590:  dbra.w    d2,dc_del
591: ; Verwaltungsdaten vor den Cachesektoren löschen
592:  move.l    cache,a0        ; Adresse Cachesektoren
593:  move.w    maxsec,d1       ; Anzahl Sektoren
594:  subq.w    #1,d1           ; minus 1
595:  move.l    #$FFFF,d2       ; d2 vorbelegen
596:  cmp.b     #2,d3           ; Drive 2?
597:  beq.s     drive_2         ; ja, Daten löschen
598: drive_1:
599:  btst      #1,3(a0)        ; Cachedaten Drive 1?
600:  bne.s     d1_nxt          ; nein, weiter
601:  move.l    d2,(a0)         ; Statistik=0,
602: ;                           rel. Statistikadresse=-1
603: d1_nxt:
604:  lea       516(a0),a0      ; nächster Sektor
605:  dbra.w    d1,drive_1
606:  bra.s     dc_end
607: drive_2:
608:  btst      #1,3(a0)        ; Cachedaten Drive 2?
609:  beq.s     d2_nxt          ; nein, weiter
610:  move.l    d2,(a0)         ; Statistik=0,
611: ;                           rel. Statistikadresse=-1
612: d2_nxt:
613:  lea       516(a0),a0      ; nächster Sektor
614:  dbra.w    d1,drive_2
615: dc_end:
616:  rts
617: ;
618: cache_del:                  ; Alle Cachedaten
                                  löschen
619:  tst.w     maxsec          ; Cachegrösse=0?
620:  beq.s     cd_end          ; dann Abbruch
621:  movem.l   d0/d7/a1,-(sp)
622:  move.l    stat,a1         ; Statistikadresse
623:  move.w    #1217,d7        ; 4872 Statistiksektoren
624: ; 87 Tracks * 14 Sektoren * 2 Drives * 2 Seiten
625: cd_del:                     ; Statistik löschen
626:  clr.l     (a1)+           ; (4 Bytes = 4 Sektoren)
627:  dbra.w    d7,cd_del
628:  move.l    cache,a1        ; Adresse Cachesektoren
629:  move.w    maxsec,d7       ; Anzahl Cachesektoren
630:  subq.w    #1,d7           ; minus 1 (für Schleife)
631:  move.l    #$FFFF,d0       ; d0 vorbelegen
632: ccloop:
633:  move.l    d0,(a1)         ; Daten löschen
634:  lea       516(a1),a1      ; nächster Sektor
635:  dbra.w    d7,ccloop
636:  clr.b     min_stat        ; Statistikminimum auf 0
637:  movem.l   (sp)+,d0/d7/a1
638: cd_end:
639:  rts
640: ;
641: ; bei Überlauf eines Statistikwertes alle
642: ; Statistikwerte wieder auf 1 setzen
643: stat_reset:
644:  bpl.s     sr_end          ; Statistikwert positiv?
645: ;                           dann Abbruch
646:  tst.w     maxsec          ; Cachegrösse=0?
647:  beq.s     sr_end          ; dann Abbruch
648:  movem.l   d0-d1/a0-a1,-(sp)
649:  move.l    stat,a0         ; Statistikadresse
650:  move.l    cache,a1        ; Adresse Cachesektoren
651:  move.w    maxsec,d0       ; Anzahl Cachesektoren
652:  subq.w    #1,d0           ; minus 1 (für Schleife)
653: sr_loop:
```

→

```
654:   tst.w      (a1)           ; Sektor nicht belegt?
655:   beq.s      sr_nxt         ; dann nächster Sektor
656:   move.w     #$100,(a1)+    ; Statistikwert auf 1
657:   move.w     (a1)+,d1       ; rel. Statistikadresse
658:   move.b     #1,(a0,d1.w)   ; Tabellenwert=1
659:  sr_nxt:
660:   dbra.w     d0,sr_loop     ; nächster Sektor
661:   movem.l    (sp)+,d0-d1/a0-a1
662:   clr.b      min_stat       ; Statistikminimum auf 0
663:  sr_end:
664:   rts
665:  ;
666:  floprd:
667:   move.l     (sp),retfloprd ; Returnadresse merken
668:   move.w     12(sp),drive   ; Laufwerks-Nr. holen
669:   bsr        cache_out      ; Sektor(en aus Cache
670:   tst.w      d0             ; alles eingelesen?
671:   beq.s      flprdend       ; ja, dann Ende
672:   move.l     #frd_nxt,(sp)  ; neue Returnadresse
673:   move.l     flp_rd,a0      ; in TOS-'floprd'-
                                  Routine
674:   jmp        (a0)           ; springen
675:  frd_nxt:
676:   move.l     retfloprd,-(sp) ; alte Returnadresse
677:   bsr        cache_in       ; nein,Sektor(en
                                  einlesen
678:  flprdend:
679:   rts
680:  ;
681:  flopwr:
682:   move.l     (sp),retflopwr ; Returnadresse merken
683:   move.w     12(sp),d0      ; Laufwerks-Nr. holen
684:   move.w     d0,drive
685:   clr.w      $444           ; '_fverify' löschen
686:   move.l     rsrcaddr,a0
687:   cmp.w      #$11,418(a0)   ; Verify an?
688:   seq        $444           ; dann '_fverify' setzen
689:   move.l     #fwr_nxt,(sp)  ; neue Returnadresse
690:   move.l     flp_wr,a0      ; in TOS-'flopwr'-
                                  Routine
691:   jmp        (a0)           ; springen
692:  fwr_nxt:
693:   move.l     retflopwr,-(sp) ; alte Returnadresse
694:   bsr        cache2in       ; Cachesektoren löschen
695:  flpwrend:
696:   rts
697:  ;
698:   dc.b 'XBRA','ST-C'        ; "XBRA"-Kennung
699:  old_mediach:
700:   dc.l 0                    ; alte 'mediach'-Adresse
701:  mediach:
702:   move.l     (sp),retmediach ; Returnadresse
                                   merken
703:   move.l     #mdc_nxt,(sp)  ; neue Returnadresse
704:   move.l     old_mediach,a0 ; TOS-'mediach'
705:   jmp        (a0)           ; anspringen
706:  mdc_nxt:
707:   move.l     retmediach,-(sp) ; alte Returnadresse
708:   cmp.w      #1,d0          ; Diskette gewechselt?
709:   ble.s      mdc_end        ; nein, Ende
710:   move.w     4(sp),d3       ; Drive-Nr. holen
711:   bra.l      disc_changed   ; Cachedaten löschen
712:  mdc_end:
713:   rts
714:  ;
715:   dc.b 'XBRA','ST-C'        ; "XBRA"-Kennung
716:  old_rwabs:
717:   dc.l 0                    ; alte 'rwabs'-Adresse
718:  rwabs:
719:   move.l     (sp),retrwabs  ; Returnadresse merken
720:   move.l     #rwabs0,(sp)   ; neue Returnadresse
721:   move.l     buffer,a0
722:   add.w      rw_abs,a0      ; in TOS-'rwabs'-Routine
723:   jmp        (a0)           ; springen
724:  rwabs0:
725:   move.l     retrwabs,-(sp) ; alte Returnadresse
726:   cmp.l      #-15,d0        ; Fehler 'unbek. Gerät'?
727:   bne.s      rwabs1         ; nein, weiter
728:   move.l     old_rwabs,a0   ; alte Routine
                                  anspringen
729:   jmp        (a0)
730:  rwabs1:
731:   move.w     14(sp),d3      ; Drive-Nr. nach d3
732:   cmp.w      #-14,d0        ; Diskette gewechselt?
733:   beq.l      disc_changed   ; ja, Cachedaten löschen
734:   rts
735:  ;
736:  ; Einige GEM-Aufrufe:
737:  form_center:
738:   move.w     #54,contrl
739:   move.l     #5,contrl+2
740:   move.l     #$10000,contrl+6
741:   move.l     rsrcaddr,addrin
742:   bsr        aes
743:   move.l     intout+2,size
744:   move.l     intout+6,size+4
745:   rts
746:  form_dial:
747:   move.w     #51,contrl
748:   move.l     #$90001,contrl+2
749:   move.l     #$10000,contrl+6
750:   move.w     d0,intin
751:   clr.l      intin+2
752:   clr.l      intin+6
753:   move.l     size,intin+10
754:   move.l     size+4,intin+14
755:   bra.s      aes
756:  form_do:
757:   move.w     #50,contrl
758:   move.l     #$10001,contrl+2
759:   move.l     #$10000,contrl+6
760:   move.w     #9,intin
761:   move.l     rsrcaddr,addrin
762:   bra.s      aes
763:  form_alert:
764:   move.w     #52,contrl
765:   move.l     #$10001,contrl+2
766:   move.l     #$10000,contrl+6
767:   move.w     #1,intin
768:   move.l     a0,addrin
769:  aes:
770:   move.l     #aespb,d1
771:   move.l     #$C8,d0
772:   trap       #2
773:   rts
774:  obj_draw:
775:   move.w     #42,contrl
776:   move.l     #$60001,contrl+2
777:   move.l     #$10000,contrl+6
778:   move.l     #4,intin
779:   move.l     size,intin+4
780:   move.l     size+4,intin+8
781:   move.l     rsrcaddr,addrin
782:   bra.s      aes
783:  evnt_mesag:
784:   move.w     #23,contrl
785:   move.l     #1,contrl+2
786:   move.l     #$10000,contrl+6
787:   move.l     #ev_buff,addrin
788:   bra.s      aes
789:  menu_register:
790:   move.w     #35,contrl
791:   move.l     #$10001,contrl+2
792:   move.l     #$10000,contrl+6
793:   move.w     ap_id,intin
794:   move.l     #acc_name,addrin
795:   bsr        aes
796:   move.w     intout,acc_id ; Accessory-Nr. merken
797:   rts
798:  rsrc_load:
799:   move.w     #110,contrl
800:   move.l     #$1,contrl+2
801:   move.l     #$10000,contrl+6
802:   move.l     #rsrcname,addrin
803:   bsr        aes
804:   tst.w      intout          ; Fehler?
805:   bne.s      rsrclend        ; nein, Ende
806:   move.b     #4+48,fehler    ; Fehlernummer
807:  rsrclend:
808:   rts
809:  rsrc_gaddr:
810:   move.w     #112,contrl
811:   move.l     #$20001,contrl+2
812:   move.l     #$1,contrl+6
813:   move.w     d0,intin
814:   move.w     d1,intin+2
815:   bsr        aes
816:   tst.w      intout          ; Fehler?
817:   bne.s      rg_end          ; nein, Ende
818:   move.b     #5+48,fehler    ; Fehlernummer
819:  rg_end:
820:   rts
```

→

```
821:  appl_init:
822:   move.w    #10,contrl
823:   move.l    #1,contrl+2
824:   clr.l     contrl+6
825:   bsr       aes
826:   move.w    intout,ap_id
827:   rts
828:  ;
829:  retfloprd  dc.l 0
830:  retflopwr  dc.l 0
831:  retmediach dc.l 0
832:  retrwabs   dc.l 0
833:  rsrcaddr dc.l 0
834:  pd         dc.l $602C
835:  sysbase  dc.l 0
836:  tos_adr  dc.w 5  ; Anzahl der Adressblöcke - 1
837:  adr_1_0  dc.l $02061986,$FC0000
838:           dc.w $748,$1B8F,$10D2,$1260
839:           dc.w $13C8,$1232,$1296,$7F0
840:  adr_1_0d dc.l $02061986,$6100
841:           dc.w $4D6,$191D,$E60,$FEE
842:           dc.w $1156,$FC0,$1024,$57E
843:  adr_1_2  dc.l $04221987,$FC0000
844:           dc.w $7F2,$1D8F,$12B6,$1444
845:           dc.w $15AC,$1416,$147A,$89A
846:  adr_1_4  dc.l $04061989,$FC0000
847:           dc.w $7C4,$1517,$1A24,$1BAE
848:           dc.w $1CF4,$1B82,$1BE2,$86C
849:  adr_1_6  dc.l $06191989,$E00000
850:           dc.w $862,$15DB,$1AE8,$1C72
851:           dc.w $1DB8,$1C46,$1CA6,$922
852:  adr_1_6n dc.l $07291989,$E00000
853:           dc.w $93A,$16DF,$1BFC,$1D86
854:           dc.w $1ECC,$1D5A,$1DBA,$9FA
855:  trap14   dc.w 0
856:  flp_rd   dc.l 0
857:  flp_wr   dc.l 0
858:  rw_abs   dc.w 0
859:  f_load   dc.w 0
860:  drive    dc.w 0
861:  oldmxsec dc.w -1
862:  maxsec   dc.w 0
863:  sperre   dc.w 3
864:  laufwerk dc.w 0
865:  ap_id    dc.w 0
866:  acc_id   dc.w -1
867:  acc_name dc.b '  ST-CACHE ',0
868:  f_name   dc.b '\STC_*.*',0
869:  rsrcname dc.b '\ST_CACHE.RSC',0
870:  err_txt  dc.b 91,'1]'
871:           dc.b 91,'ST-CACHE Version 2.71| |'
872:           dc.b 'Fehler #'
873:  fehler   dc.b 0
874:           dc.b ' bei
                 Installation|aufgetreten...]'
875:           dc.b 91,'  Ok  ]',0
876:  write    dc.b 0
877:  min_stat dc.b 0
878:  ;
879:  ; * Voreinstellung (Format wie Steuerdatei-
                                         Name);
880:  filename dc.b 'STC_ABVF.128'
881:  ;
882:  size     dc.l 0,0
883:  buffer   dc.l 0
884:  stat       dc.l 0
885:  cache      dc.l 0
886:  aespb      dc.l contrl,global,intin
887:             dc.l intout,addrin,addrout
888:  contrl     ds.w 11
889:  global     ds.w 15
890:  intin      ds.w 128
891:  intout     ds.w 128
892:  addrin     ds.w 128
893:  addrout    ds.w 128
894:  ev_buff    ds.l 4
895:  dta        ds.b 44
896:             ds.l 12
897:  regsave    ds.b $100
898:  stack:
899:   END
```

```
 1:  ' ***********************************************
 2:  ' *                ST-CACHE                     *
 3:  ' *   BASIC-Loader zu Erstellung des RSC-Files  *
 4:  ' *   von Matthias Andrä   (c) MAXON Computer   *
 5:  ' ***********************************************
 6:  '
 7:  Open "O",#1,"\ST_CACHE.RSC"
 8:  For I=1 To 364
 9:    Read A$
10:    A=A+Val("&H"+A$)
11:    Print #1,Mki$(Val("&H"+A$));
12:  Next I
13:  Close #1
14:  If A<>3473201
15:    Print "Fehler in Data-Zeilen !"
16:    Kill "\ST_CACHE.RSC"
17:  Endif
18:  End
19:  '
20:  Data 0,F4,A0,A0,A0,0,24,A0,0,2D4,14,1,3,0,0,0,0
21:  Data 2D8,5370,6572,7265,3A00,4772,949E,653A,41
22:  Data 3A00,4361,6368,653A,41,3A00,423A,42,3A00
23:  Data 3132,3800,5F5F,5F00,3939,3900,4B62,7974
24:  Data 6500,5354,2D43,4143,4845,56,5B32,2E37,315D,
        0,30,352F,3930,2C20,204D,6174,7468,6961
25:  Data 7320,416E,6472,8400,0,4641,5354,4C4F,4144
26:  Data 56,4552,4946,5900,4C99,5343,4845,4E00,4F4B
27:  Data 0,046,0,4A,0,4E,3,6,0,1180,0,FFFE,4,4,0,61
28:  Data 0,69,0,6A,5,6,0,1180,0,FFFF,8,1,0,6B,0,82
29:  Data 0,83,5,6,2,1180,0,FFFF,17,1,FFFF,1,13,14,0
30:  Data 10,3,1122,0,0,25,E,B,2,A,14,0,10,2,1143,2
31:  Data 6,15,7,3,FFFF,FFFF,1C,0,0,0,24,2,1,7,1,4,
        FFFF,FFFF,1C,0,0,0,2C,3,5,6,1,5,FFFF,FFFF
32:  Data 1A,1,10,0,33,A,1,4,1,6,FFFF,FFFF,1C,0,0,0
33:  Data 36,3,3,6,1,7,FFFF,FFFF,1A,1,10,0,3D,A,3,
34:  Data 4,1,8,FFFF,FFFF,1A,1,10,0,40,F,3,4,1,9,
35:  Data FFFF,FFFF,1A,1,10,0,43,F,1,4,1,A,FFFF,FFFF
36:  Data 1E,8,10,0,A0,A,5,9,1,1,FFFF,FFFF,1C,0,0,0
37:  Data 52,E,5,5,1,F,C,E,14,0,0,FD,1101,2,1,1A,4,D
38:  Data FFFF,FFFF,1C,0,10,0,58,9,1,8,1,E,FFFF,
        FFFF,15,0,0,0,BC,13,2,205,600,B,FFFF,FFFF
39:  Data 15,0,0,0,D8,5,3,410,600,12,10,11,14,0,10,2
40:  Data 1143,19,8,A,5,11,FFFF,FFFF,1A,1,10,0,84,1,1,
41:  Data 8,1,F,FFFF,FFFF,1A,1,10,0,8D,1,3,8,1,13,
42:  Data FFFF,FFFF,1A,5,30,0,94,1A,6,8,1,0,FFFF,FFFF,
43:  Data 1A,27,30,0,9C,1E,1,4,2,0,F4
```

ATARI

1040 STE / SM 124	DM	1348,–
Mega ST 1 / SM 124	DM	1348,–
Mega ST 2 / SM 124	DM	2128,–
Mega ST 4 / SM 124	DM	3098,–
Megafile 30	DM	848,–
Megafile 60	DM	1178,–
Megafile 44 incl. Cartridge	DM	1998,–
Laser SLM 804	DM	2348,–
SM 194 19" Monitor für ST's	DM	3748,–

Alle Preise incl. Mwst.. Wir führen nur deutsche Originalware direkt vom autorisierten ATARI - Händler und geben Ihnen ein volles Jahr Garantie!

KRÜGER EDV - MARKETING

Tel. 0 28 57 / 17 01 Fax. 0 28 57 / 17 00

Rees ☆ Düsseldorf ☆ Viersen

ROMAN MODERN

Komplette und z.Z. umfangreichste Schriftfamilie für **Signum** in fünf harmonisch aufeinander abgestimmten Schriftschnitten für 24-Nadel- oder Laser-Drucker:

Roman Modern Regular
Roman Modern Bold
Roman Modern Italic
Roman Modern Bold Italic
ROMAN MODERN CAPS

Jeder Schnitt liegt in sieben Größen vor (6, 8, 10, 11, 12, 14, 16pt), verfügt über Ligaturen, Sonderzeichen und einen Grundbestand an akzentuierten Buchstaben — je Schnitt und Größe mehr als 170 Zeichen.

Als Vorlage diente die TEX-Schriftfamilie CMR, so daß nun auch Signum-Anwender eine ähnlich einzigartige typografische Ausgabequalität erzielen können.

Roman Modern komplett 130,– DM

(34 Fonts, je Font ein Standard- und ein Sonderzeichensatz, Macro's, ausführliche Dokumentation)

Gegen Verrechnungsscheck oder per Nachnahme, zzgl. 5,– DM Versandkosten bei *

Detaillierte Informationen und Schriftproben gegen 2,– DM Rück-Porto (in Briefmarken) bei *

* H.Schlicht, Ketzendorfer Weg 4H,
2104 Hamburg 92, Tel.: 040 / 7 01 64 92

SW-HAUSHALT V 1.0

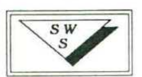
Die private Haushaltsbuchführung für jedermann. Einfache Bedienung. Graf. Auswertungen. Monats- u. Jahresabschuß auf Drucker u. Monitor. Verwaltung von Schecks aller Art. Suchen, Ändern u. Löschen aller Buchungen. Deutsches Handbuch. Voll unter GEM nur DM 69

SW-VIDEO

Verwaltet 2500 Videofilme nach 24 verschiedenen Filmarten (Action, Erotik). Ermittelt alle Restlaufzeiten. Komfortables Löschen und Ändern aller Daten. Sehr schnell bei allen Such- u. Sortierfunktionen. Alle Ausgaben auf Drucker u. Monitor. Etikettendruck. Deutsches Handbuch. Voll unter GEM nur DM 89

Porto: VK 4,- / NN 6,-

Sie erhalten bei uns über 1000 PD-Disketten. Katalog kostenlos.



SW-Software
Beethovenstr. 10
7938 Oberdischingen
Tel. 07308/8325

Ausgewählte PD-Software

Computer & Electronic & Zubehör HERGES

Obere Rischbachstraße 88 • 6670 St. Ingbert
Telefon (06894) 38 31 78 / Telefax (06894) 38 28 55

Atari-Computer + Zubehör:

Mega ST-2 komplett mit Zubehör (ohne Monitor)	DM 1998,-
Mega ST-2 wie oben, mit 4MByte RAM aufger. (ohne Monitor)	DM 2658,-
Atari Mega-ST mit PC-Speed eingebaut Aufpreis	DM 500,-
Megafile 30, komplett mit Zubehör	DM 888,-
Megafile 44, komplett mit Zubehör und 1 * Cartridge	DM 2128,-
Wechselcartridge (SQ 400/44 MB) für Megafile 44	DM 268,-
Laserdrucker SLM-804, komplett mit Zubehör	DM 2698,-
Toner-Cartridge für Laserdrucker SLM-804	DM 95,-

Soundchip	25,-	Glue 147	147,-	MFP	29,-	Shifter	168,-
WDc 1772-2	39,-	MMU	147,-	DMA	147,-	Blitter	188,-
Tast.Proz.	69,-	68000	21,-	6850	4,-	RP5-C15	28,-
TL-497ACN	8,35	TL-7705	3,90	PC900	4,65	74LS/usw	a.A.

Dyn. RAMs + Gals + Gal-Zubehör:

DRAM 1 MB* 1, Typ 51 1000-80 ns (andere xxx-ns a.A.)	DM 22,95
DRAM 256 KB* 1, Typ 41 256 100 ns (andere xxx-ns a.A.)	DM 7,35
2 MB RAM-Erweiterung (0 MB bestückt) auf max. 2.5/ 4 MB	DM 128,-
2 MB RAM-Erweiterung (2 MB bestückt) auf max. 2.5/ 4 MB	DM 538,-
Gal 16V8-25L (Sockel für Gal 16-V8-25L DM -,35)	DM 4,65
Gal 20V8-25L (Sockel für Gal 20-V8-25L DM -,40)	DM 6,25
Gal-Programmer für Gals 16V8 und 20V8, incl. Software, ab	DM 138,-
Gal-Assembler für die Erstellung der Fusematrix	DM 148,-

Eproms + Eprommer + Platinen:

E-Prom 27C256-200 ns (32 KB*8) Programmiersp. 12.5 V.	DM 7,90
E-Prom 27C512-200 ns (64 KB*8) Programmiersp. 12.5 V.	DM 13,70
E-Prom 27C010-200 ns (128 KB*8) Programmiersp. 12.5 V.	DM 31,60
Epromdisk 512 KB kompl. mit Software, 0 KB bestückt	DM 158,-
Epromdisk-Erweiterung (auf max. 1 MB) 0 KB bestückt	DM 88,-
ST-Eprommer mit Software, bis Eproms 27011/27C011	DM 168,-

IC-Sockel LC-Low Cost / PZ-Präz. / SM=schmal:

* 64pol. PZ=6.75	* 48pol. PZ=2.95	* 40pol. LC=0.75	* 40pol. PZ=2,25
* 32pol. LC=0.55	* 28pol. LC=0.45	* 24pol. LC=0.42	* 24pol. SM=0,40
* 20pol. LC=0.35	* 18pol. LC=0.30	* 16pol. LC=0.25	* 14pol. LC=0,20
* 08pol. LC=0.15	* Einzelkontakt-PZ=0.07		* Stecker- PZ=0,10

Sonstiges aller Art:

Schaltnetzteil-Platine in Industriequalität.....!!!
Kompl. aufgebaut + geprüft, kurzschlußfest, superleicht, dyn. Stromentnahme bis 60 W, VDE 0804/0806 TÜV Rheinland, Abm. 160x100x45 mm,

Typ + 5V/5 Amp., + 12 V/2.5 Amp., -12 V/0.5 Amp.	DM 67,-
Star LC-24/10, 7 KB-Puffer, Paper Park-Funktion	DM 728,-
Quarz-Oszillator 10.24 MHz oder 16 MHz, bitte angeben	DM 8,-
PC-Speed (MS-DOS auf dem Atari ST)	DM 448,-
Mr.Robot - Actionspiel mit 18 Levels	DM 39,-

Auszug aus unserem Lieferprogramm, bitte Komplettliste anfordern!
Lieferung zuzügl. Porto/Verp. per Post! Alle Angebote freibleibend!
***** Bei Terminvereinbarung auch Selbstabholung *****

ALLES FÜR ATARI ST

SOFT 2000

Preiswert – Qualität – Service – Modem – Neu

SPEICHERAUFRÜSTUNG für alle ATARI's

260/520ST	auf 2,5MB (lötfrei)	DM 696,-
1040ST	auf 2,5MB (lötfrei)	DM 696,-
1040STE	auf 2,0MB (lötfrei)	DM 496,-
TOS 1.4	original ATARI	DM 194,-
A COPY ST		DM 65,-
GFA-Entwicklungssystem 2.02		DM 47,-

NEU – VORTEX DATAJET – NEU

Festplatte	X30/30MB HDS	DM 1298,-
Wechselplatte	R44/44MB RHDS	DM 2498,-
Fest/Wechselplatte	X40R44/90MB CDS	DM 3898,-

PUBLIC-DOMAIN-SOFTWARE ab DM 5,-
pro Disk, z.B.aus ST-Computer/PD-Pool/PD-Journal

KATALOG + PD-LISTE auf DISK
kostenlos, lediglich für die Versandkosten bitten wir um Zusendung von DM 3,50 in Briefmarken.

Ab DM 100,- Warenwert liefern wir frei Haus.
SOFT aus 2000 schickt Ihnen gern Info's.

Computer Soft- & Hardware Tel. 040/6556496
Postfach 740162 040/6905646
2000 Hamburg 74 Btx 040/6514966

Atari-ST Speichererweiterung

(inklusive Einbau, 14% MwSt und Versand!)

260-ST oder 520-ST	auf 1 MByte	190,- DM
	auf 2,5 MByte	520,- DM
	auf 4 MByte	1000,- DM
520-ST+	auf 2,5 MByte	550,- DM
	auf 4 MByte	1000,- DM
1040-STF	auf 2,5 MByte	550,- DM
MEGA-ST2	auf 4 MByte	520,- DM

Einbau von Echtzeituhr in 260/520-ST:	60,- DM
Umbau von 2 auf 6 Epromsockel:	55,- DM
6 Stück EPROM 27C256:	45,- DM
Disklaufwerk 5,25" (ohne Gehäuse)	160,- DM
Disklaufwerk 3,5" (ohne Gehäuse)	160,- DM
Disklaufwerk 3,5" (anschlußfertig)	235,- DM

CHRISTIAN RUPP
Am Kronwerk 9 6740 Landau
Tel. 06341/84993

WARUM denn immer so kompliziert ?

Nehmen Sie doch PegaSoft-Programme!
Die sind schnell und einfach -
weil nichts Überflüssiges stört !

PegaSoft

PegaFAKT ADRESSEN •Anzahl u. Rg-Summe der Einkäufe •Datum letzter Einkauf •6 Rabattgruppen •Suchen in allen Feldern LAGER •Sollbestand •Verkaufte Menge •6 Rabattgruppen •Suchen in allen Feldern •Unterbestandslisten •Automatische Preiskalkulation FAKTURIERUNG •Rg-Nr-Routine •alle Rg-Artikel (max. 50) werden gleichzeitig am Monitor angezeigt - Ändern, Einfügen und Löschen ist jederzeit möglich •Formular einstellbar (auch DIN A5) •Brutto- oder Nettopreise (USt-Ausdruck erfolgt entsprechend) •Lieferscheine mit und ohne Preise •Versandaufkleber mit großer Postleitzahl und NN-Betrag •Rg abspeichern, neu einladen oder anfügen •Off.Posten-Liste •Auslandsanpassung (Währung, Adresse rechts) •Daten im ASCII-Format selektiert exportierbar •eingeb. Editor für Listen u. Etiketten •Alle PRG-Teile stehen gleichzeitig im Speicher und können mit Funktionstasten direkt aufgerufen werden - unglaubl. schnell und bequem! •Konvertierprogramm für PD-Version •Tastaturschablone •1-seitige Diskette (DEMO 10,- DM) **99,–**

PegaSTIC Universelles Etikettendruckprogramm •alle Endlosformulare bis 6 Bahnen •alle Druckerschriften und -zeichen (z.B. Striche, mathem. Zeichen,...) ansprechbar •auf alle Drucker und Etikettengrößen anpaßbar •Aufruf von bis zu 48 verschied. Schriften durch Eingabe einer Zahl •Autom. Numerierung mögl. (Startwert u. Schrittweite wählbar) •Beliebiger Wiederholungsdruck •Texteditor •Anzahl Zeilen/Etikett beliebig (automat. Zentrierung = gleichgroße Zeilenabstände im Etikett) •Etiketten speichern u. konvertieren in andere Größen •Adressen u. Artikel aus PegaFAKT einbindbar •incl. Zeilenlineal •1-seitige Diskette **49,–**

Schweiz: Pierre Scherz Postfach 17 9542 Münchwilen (073) 263277

PegaSoft Rudolf Gärtig Ringstr.4 D-7450 Hechingen 11 (07477)8158
Versandkosten: Vorkasse 3,50/NN 6,- Händleranfragen erwünscht!
Fordern Sie auch unsere kostenlose Computer-Zubehörliste an !
PD für nur 4,- DM (ST-Comp./XEST/GFA/PD-Pool/...) Liste 2,-

› ATARI ST ‹

Anwendersoftware		Spielesoftware	
CCD		F-16 Combat Pilot	69,-
ST Pascal + V2.0x	220,-	Gunship	79,-
Tempus Editor 2.0	100,-	Indiana Jones (Adv.)	69.-
Tempus Word	a.A.	Oil Imperium	53,-
Assembler Tutorial	90,-	Populous	69,-
GFA		Rick Dangerous	69,-
GFA Basic 3.0 (I.+C.)	180,-	RVF Honda	69,-
GFAAssembler	140,-	Sleeping Gods Lie	69,-
Omikron		Spherical	53,-
Omikron Basic Comp.	170,-	Zak McKraken	69,-
Mortimer, Utility	75,-	Star Trash	50,-
Appl. Syst.		Diskbox 3,5" 80er	19,-
Signum!2	418,-	Supercharger	690,-
Scarabus	90,-		
Signum Revers Acc.	90,-	Traktrix	80,-
Protos	64,-	Approximationsprg.	
Fontdisketten verfügbar		für sämtl. Fkttypen	

Porto: Vorkasse 4,- Nachnahme 7,- DM

Computerversand G. Thobe
Pf 1303 - 4570 Quakenbrück
Tel.: (05431) 5251

Professionelle Schön-Schrift

mit **Signum** und **Laser-/24-Nadeldrucker**
→ jetzt auch für Textverarbeitung **script**

- Modernes, optimal lesbares Schriftbild
- in den Größen 8, 10, 12 und 15 Punkt,
- für Laserdrucker auch 6 und 20 Punkt,
- einschließlich unproportionaler Ziffern
- normal und fett, und dazu ein schlau
- ausgetüftelter „SONDER"-Zeichensatz

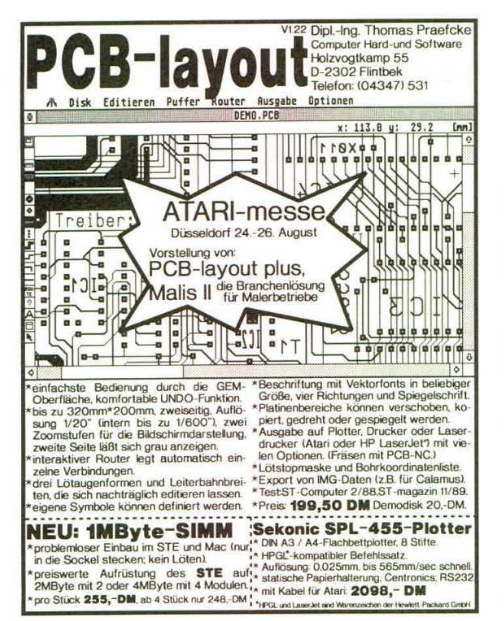
mit griechischem Alphabet (A, α, B, β, Γ, γ, Δ, δ, ..., Ω, ω), römischen Zahlen (I, II, III, IV, ... X, ... MCMLXXXIX), fremdsprachigen (Å, å, Ç, ç, Ø, ø, Œ, œ, ñ, ¿), kaufmännischen (£, ‰, ®, ©, ⓦ) und mathematischen Zeichen (∞, ∫, ∫, ≡, ≠, ≢, <, ≦, ≧, >, ~, ≈, ≅, ±, ×, ÷, ∢ usw.) und vielem mehr (①, ... ⑨, ⑩, ... ㊿, ϟ, ↗, →, ←, ↑, ↓) für die unterschiedlichsten Einsatzbereiche.

Ausführliche Info mit Schriftprobe
■ für 3,- DM in Briefmarken anzufordern bei:
Walter Schön, Berg-am-Laim-Str. 133a, 8000 München 80, Tel. (0 89) 4 36 22 31.

Neu für Laserdrucker: senkrecht laufende 10-Punkt-Schrift



Elektronik Software
ASPICE
Industriell einsetzbarer Transistor-Analogsimulator mit bester GEM-Grafik für Atari ST
Optional: Halbleiter-Modellbibliothek.
Spezialversion auch für 68000/68881 und 68020/68881
Preis zuzüglich Porto und Verpackung
ab DM 348.-
ADIGSI
neu
Industriell einsetzbarer Worst-Case Digitalsimulator mit Schaltungseingabe für Atari ST
Optional: 74LSxx-Bauteile-Bibliothek.
Spezialversion auch für 68000/68881 und 68020/68881
Preis zuzüglich Porto und Verpackung
ab DM 348.-
CAE SYSTEME RUFF Dipl.-Ing.
Tel. 07307-24187 FAX -23209
D-7910 Neu-Ulm Pf. 1942
professionelle Elektronik - Simulatoren für alle Atari ST
alle Atari ST ab 1 MByte RAM erfordert Joystickport für Kopierschutz



PCB-layout
V1.22 Dipl.-Ing. Thomas Praefcke
Computer Hard-und Software
Holzvogtkamp 55
D-2302 Flintbek
Telefon: (04347) 531
ATARI-messe
Düsseldorf 24.-26. August
Vorstellung von:
PCB-layout plus,
Malis II die Branchenlösung für Malerbetriebe
*einfachste Bedienung durch die GEM-Oberfläche, komfortable UNDO-Funktion.
*bis zu 320mm*200mm, zweiseitig, Auflösung 1/20" (intern bis zu 1/600"), zwei Zoomstufen für die Bildschirmdarstellung, zweite Seite läßt sich grau anzeigen.
*interaktiver Router legt automatisch einzelne Verbindungen.
*drei Lötaugenformen und Leiterbahnbreiten, die sich nachträglich editieren lassen.
*eigene Symbole können definiert werden.
*Beschriftung mit Vektorfonts in beliebiger Größe, vier Richtungen und Spiegelschrift.
*Platinenbereiche können verschoben, kopiert, gedreht oder gespiegelt werden.
Ausgabe auf Plotter, Drucker oder Laserdrucker (Atari oder HP LaserJet) mit vielen Optionen. (Fräsen mit PCB-NC.)
*Lötstopmaske und Bohrkoordinatenliste.
*Export von IMG-Daten (z.B. für Calamus).
*Test:ST-Computer 2/88,ST-magazin 11/89.
*Preis: 199,50 DM Demodisk 20,-DM.
NEU: 1MByte-SIMM
*problemloser Einbau im STE und Mac (nur in die Sockel stecken; kein Löten).
*preiswerte Aufrüstung des STE auf 2MByte mit 2 oder 4MByte mit 4 Modulen.
*pro Stück 255,-DM, ab 4 Stück nur 248,-DM
Sekonic SPL-455-Plotter
* DIN A3 / A4-Flachbettplotter, 8 Stifte.
* HPGL*-kompatibler Befehlssatz.
* Auflösung: 0,025mm, bis 565mm/sec schnell.
* statische Papierhalterung, Centronics; RS232
* mit Kabel für Atari: 2098,- DM
*HPGL und LaserJet sind Warenzeichen der Hewlett-Packard GmbH



DRUCKSACHEN
ATARI
software - hardware
scann-service - laserdruck-service
verkauf - versand - beratung
drucke nach ihren vorstellungen!
STAD 1.3+ 159.--
Draw! 3.0 119.--
* Sherlock 3.0 899.--
GFA-Basic 3.5 Int.+Comp. 259.--
NeoDesk 2.05 77.--
Scann-Service ab 2.20
Laserdruck-Service ab 1.60
* Lern ST plus 55.--
* Presto 888.--
Creator 1.1 229.--
Writer ST 1.4 139.--
AS Sound Sampler III 578.--
SuperCharger 739.--
(* Demo-Disk geg. 10 DM Schein)
DRUCK- u. COMPUTERKATALOG anfordern!
Marcus Damme - Druck & Computer
Grambeker Weg 40 - 2410 Mölln
Tel.: 04542/87258 --> 16.00-22.00 Uhr



FiBu++
Das Finanzbuchführungsprogramm
Bilanz und
Gewinn- und Verlustrechnung
oder Einnahmeüberschußrechnung
NEU
Version 2.0 mit komfortabler Debitoren- / Kreditorenverwaltung
Die professionelle Softwarelösung für Gewerbe und Freiberufler
Einige Leistungsmerkmale:
Vorbildliche Ergonomie, umfangreiche Hilfen durch On-Line-Handbuch und 'handfestes' Handbuch. Sehr schnelles Buchen und Auswerten. Ablehnung falscher und unsinniger Buchungen. Umsatzsteuerberechnungen. Offene Posten mit automatischem Ausbuchen, Skontieren und vielfältigen Auswahl-, Sortier- und Statistikmöglichkeiten. Uptodate-Service.
Für Atari ST, s/w
Dipl.-Ing. W. Scheidt
Espanstr. 76
D-8510 Fürth
(0911) 791448
Preise (incl. Handbuch)
FiBu++ 498.-
Demo 40.-
(wird angerechnet)



CSH SPECIALs
PREISSENKUNG
SUPRA Modem 2400RS DM 298,00
SUPRA Modem 2400PC DM 228,00
SUPRA SCSI-Interface ab DM 198,00
SGSNet MIDI-Netzwerk DM 398,00
TURBO16 16MHz Beschleuniger DM 548,00
Festplatten [SCSI] ab DM 898,00
Einbaufestplatten ab DM 898,00
PowerPack [WP+HD] ab DM 2398,00
Wechselplatten u. -Floppy ab DM 1398,00
ATARI MEGA 2-30 DM 3248,00
CSH Ing.Büro Dipl.Ing. M.H. Krompasky
Schillerring 19, 8751 Großwallstadt
Tel. 06022 - 24405 FAX 06022-21847



PORTRAITS UND MODEGRAPHIKEN
für DTP und Graphikanwender. Verschiedenste Einsatzmöglichkeiten, zur Werbung, Ausgestaltung von Layouts. Eigens entworfene Graphiken, gescannt mit 200 bis 400 dpi. Nicht auf Screenformat begrenzt. Je nach Bildgroße auf mehrere Screens verteilt; inclusive jeweils ein Gesamtbild. Somit auch für 600x400 Monitore nutzbar. Sehr gute Bildqualität beim Drucken. Auch bei Orginalgröße. Nutzbar auch für Anwender die nicht über Vektorgraphik verfügen (Signum!). Über 800k gepact auf Markendisketten. Einführungspreis : 60.-- DM incl. Porto und Verpackung. Zusätzlich Unterlagen zur Zusammensetzung der einzelnen Bilder. Verrechnungscheck, bar oder Überweisung. Zur Bestellung genügt eine Postkarte. Auf Wunsch fertigen wir auch spezielle Graphiken an. Weitere Grapiken in Vorbereitung.
LUKIDIS G. - KIRCHHEIMER STR. 54-1 - 7440 NÜRTINGEN
Bankverbindung: Volksbank Nürtingen Konto-Nr: 27 31 BLZ: 612 901 20



VORTEX
HD plus Festplatten
20 MB 848,-
30 MB 948,-
40 MB 1148,-
60 MB ! 1348,-
120 MB ! 1848,-
44 MB incl. Medium 2148,-
Platten- und Lüfterabschaltung !!!
Autopark, Autoboot, Cache u.v.m.
Auf Wunsch: randvoll mit PD Software
(MAXON Liste PD 140 - 333) 1 MB nur 2,-DM
tel - Soft Thomas Leschner 06421/25770
Universitätsstr. 40 3550 Marburg Fax 14425

NEU – jetzt Version 4.2!

KFZ–ST

Die KFZ–Kostenanalyse für alle ATARI–ST (sw)

- Berechnung und Dokumentation aller anfallenden Kosten rund ums Auto
- Statistische Auswertung von Verbrauch, Benzinpreis & Kosten in Diagrammen
- Terminüberwachung für TÜV/ASU/Inspektion
- Komfortable Eingabe- u. Ediermöglichkeit
- Volle GEM–Unterstützung; Bedienung über Maus und Tastatur möglich; Hilfefunktion
- Einfache Druckeranpassung
- Lieferung incl. ausführlichem Handbuch

DM 98,-
Demo 10.- (wird anger.)
Info kostenlos

Dipl.-Ing. M. Heydrich
Lange Zeile 84
8520 Erlangen
Tel. 09131/55204 ab 18 Uhr

Händleranfragen erwünscht!

80C196–ENTWICKLUNGS–SYSTEM FÜR DEN ATARI–ST

Lieferumfang:
Euro-Karte mit 80C196KB-12MHZ, 32KRam, 32kEprom mit Monitorprogramm, RS-232, Wrap-Feld ca. 100mm x 70mm

Voll GEM-gesteuertes MENU mit ASSEMBLER/LINKER für 80C196 mit Fenster für Kommunikation über RS-232 mit der Entwicklungskarte, 130 Seiten Handbuch, Hilfsprogramme

FÜR MS-DOS RECHNER IN VORBEREITUNG !!!

VERTRIEB UND INFORMATION:

HELMUT CORDES	SIEGFRIED CORDES
HOFFELDSTRAßE 18	HOCHLANDWEG 3
5060 BERG.GLADBACH 2	8153 NEUKIRCHEN
02202/56156	08020/1490

KOMPLETT: DM 449,-

LEERPLATINE, HANDBUCH MONITOR-EPROM, SOFTWARE + BAUTEILE(KOMPLETT) DM 339,-

---

WBW-Service
Willi B. Werk

********* NEU *********
MegaPlot 169,--
Der Werteplotter
** zum Einführungspreis **
Demo 10,--
********* NEU *********

Signum! Zwei	a.A.
STAD 1.3+	151,--
Script	171,--
Headline 3.0	86,--
That's Write Profi	298,--
Calamus	691,--
GFA-BASIC 3.5	216,--
GFA-ASSEMBLER	119,--
GFA-STRUKTO	206,--
GFA-DRAFT-plus	305,--
Btx/Vtx-Man. o.l.	245,--
Anti Viren Kit 3.0	79,--
Adimens ST plus	329,--
AdiTALK ST plus	329,--
THEMADAT 4.0	209,--
TiM I	119,--
BS-Handel/3	621,--
fibuMAN e	329,--
TEMPUS V.2.0x	102,--
Diskus	125,--
NeoDesk	76,--
1ST-ADDRESS	89,--
Harlekin	112,--
Mortimer	79,--
NeoDesk	76,--
MegaPaint II	396,--
Arabesque	216,--
PC-Speed V.1.4	439,--
SpeedBridge	68,--
Supercharger 1MB	739,--
Fuji MF2DD 10St.	25,--

Alle Preise in DM, inkl. der gesetzl. MWSt. Vorbehaltlich Irrtümer und Preisänderungen. Bei Vorkasse 2% Skonto, zuzügl. DM 5,50 Versandkostenanteil; bei Nachnahme kein Skonto, zuzügl. DM 9,50 Versandkostenanteil. Kein Ladenverkauf! Dies ist nur ein kleiner Ausschnitt aus unserer Preisliste, die wir Ihnen gerne kostenlos übersenden.

Sielwall 87, D-2800 Bremen 1 - Tel. 0421/75116

COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B, D-7000 Stuttgart 80
Telefon.: 07 11 / 6 78 73 92

VIDEO STAR: Neuartiges, megastarkes Videoverwaltungs- und Bearbeitungsprogramm. *Video Star* zeichnet sich durch seine vielen Eingabemöglichkeiten aus. Zum Beispiel: Nummer, Bandsorte/länge, Aufnahmemonat, Genre, Farb/SW-Film, Filmdauer, ZW-Start/Ende, Titel, Regie, Drehbuch, Kamera, Musik, Trick, Jahr d. Kopie, Schauspieler und vieles mehr. Natürlich können Sie auch eine Inhaltsangabe verewigen oder Aufkleber erstellen, oder . . . nur DM 49,--

MASTER MATRIX: MM beherrscht auch die komplexwertige Matrizenrechnung. Jeder Algorithmus ist zweimal implementiert, einmal im Reellen und zum anderen im Komplexen. MM hilft bei der Berechnung von linearen Gleichungssystemen, Determinate, charakter. Polynoms oder einer Inversen. nur DM 49,--

LISTING PRINTER: Druckerutility von S. Hübner. nur DM 15,--

HASCS - THE GAME CREATOR: Rollenspielentwicklungsset. Entwickeln Sie Ihre eigenen Fantasyspiele (im Ultima-Look) - ohne jegliche Programmierkenntnisse. Ausführliches Handbuch liegt bei! nur DM 59,--

EXPEDITION NACH VARTORIA: Fesselndes Fantasyspiel, das Sie garantiert für mehrere Tage nicht mehr aus seinem Bann läßt. nur DM 29,--

Leerdisketten: 1 Päckchen (10 Stk.) No-Name mit bunten Labels, Double Sided/ Double Density / 3.5" Inch / 135 TPI; gute Qualität nur DM 14,95

Original-Software läuft nur in hoher Auflösung, 2DD / Preise zzgl. DM 5,-- Porto! *Bitte fordern Sie unseren Gratiskatalog "Software-Impulse" an!*

MITGLIED DER INTERESSENGRUPPE "THE BETTER BIT"

Besuchen Sie uns auf der ATARI - Messe in Düsseldorf

Heim Verlag

Soft – Warehouse
/// Computer– & Mediavertrieb ///

If we don't have it, you don't need it!

Auszug aus unserem aktuellem Angebot:

Atari STE mit SM 124 + 4 Megabyte Power	2.159,--
SIM 1 MBx8 für Atari STE	219,--
SCSI-Festplatte 104 Megabyte, 19 ms Zugriffszeit	1.939,--
SCSI-Festplatte 49 Megabyte, 40 ms Zugriffszeit	1.229,--
Drucker NEC P6 plus (deutsches Gerät mit dt. Handbuch)	1.178,--
Drucker NEC P7 plus (deutsches Gerät mit dt. Handbuch)	1.498,--
Atari Lynx Superspielkonsole mit 4096 Farben, Stereosound	329,--
Games (z.B. Electrocop, Chip's Challenge, Gauntlet III Paperboy, Grid Runner, California Games ......)	ab 69,--
Panasonic Anrufbeantworter der Spitzenklasse	
- KX-T 4200* (mit Fernabfrage, Funktelefon, etc.)	519,--
- KX-T 2429* (das ultramoderne Teil, sprachgesteuerte Fernabfrage, synth. Stimme spricht Zeit&Datum d.Anrufs auf Band)	498,--
- KX-T 1430 (für Preisbewußte, mit FTZ. Nr.)	275,--
- 10 Stück Disketten 3,5 "	12,50
- 10 Stück Disketten 5,25 "	5,50
- Farbbänder (für alle gängigen Drucker & Schreibmaschinen)	ab 10,50

* Betrieb am Fernsprechnetz der DBP verboten. Nichtbeachtung kann strafrechtl. Folgen nach sich ziehen.

Soft-Warehouse, Computer- & Mediavertrieb, Inh. Oliver C. Kohl
Überfelder Str.11, 5657 Haan 1, Tel.02129-6366

---

10 PD-Disketten
aus „ST-Computer" oder PD-Pool nach freier Wahl
Komplettset nur 29,-

Katalogdiskette mit allen Diskinhalten gratis!

20 Disks nach Wahl	49,-
50 Disks nach Wahl	109,-
100 Disks nach Wahl	199,-
PD-Set A (von allem etwas)	10 Disks (2s) je DM 29,-
PD-Set B (Spiele für s/w)	
PD-Set C (Spiele für Farbe)	
PD-Set D (Anwendungen)	
PD-Set E (Utilities)	
PD-Set F (Grafik+Pics)	
Sets B-F zusammen (> 35 MByte)	99,-
(=46 DM gespart)	
Flugsimulator II nur	79,-
STAD 1.3+	144,-
Harlekin	109,-
Signum!2	333,-
Script	169,-

Alle Preise incl. Disks, Versand- und Verpackungskosten sowie unserer Katalogdiskette. Wie liefern innerhalb 24h - garantiert.

Gerald Köhler
PD-Schnellversand Atari ST
Mühlgasse 6 / 6991 Igersheim
Hotline: 07931/43922+44661

---

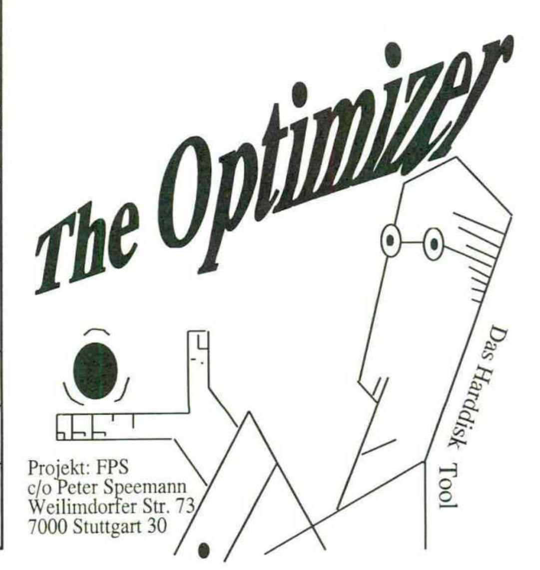
20 PD-Disketten
aus „ST-Computer" oder PD-Pool nach freier Wahl
Komplettset nur 49,-

Katalogdiskette mit allen Diskinhalten gratis!

10 Disks nach Wahl	29,-
50 Disks nach Wahl	109,-
100 Disks nach Wahl	199,-
ST-Serie (1-333, 221 Disks)	399,-
2000'er Serie (1-48, 52-140)	229,-
Signum!-Font-Set (8 Disks)	25,-
Tex-Komplettset (10 Disks)	29,-
MS-DOS-Top-PD-Set (10 Disks)	29,-
Chaos Strikes Back	59,-
Dragon Flight	88,-
Flexdisk/HD-Utility	je 49,-
GFA-Assembler	128,-
FiBu Man e/f	333,-/666,-
10 Leerdisketten 3,5' 2DD	13,90
Speichererweiterung 512 KB	179,-

Alle Preise incl. Disks, Versand- und Verpackungskosten sowie unserer Katalogdiskette. Wie liefern innerhalb 24h - garantiert.

Gerald Köhler
PD-Schnellversand Atari ST
Mühlgasse 6 / 6991 Igersheim
Hotline: 07931/43922+44661





King Richard sitzt im dunklen Verlies eines österreichischen Schlosses. Deshalb macht sich Ritter Ivanhoe auf den Weg, um seinen geliebten König zu befreien. Dazu muß er fünf gefährliche Gegenden durchqueren, mit Schild und Schwert gegen buckelige Keulenschwinger, miese Messerwerfer und heranfliegende Adler kämpfen. Kunterbunte Comicgrafiken, abwechslungsreiche Levels und tolle Musiken sorgen für gute Unterhaltung.



Ivanhoe

Mit dem Flugzeug über Feindesland. Überall schießen Panzer Salven in die Luft, ebenso wie Kanonen und Gefechtsstände. Außerdem machen feindliche Jets das Pilotenleben schwer. Da gibt's nur eins: ballern, bis der Daumen kracht. "Sonic Boom" ist prima spielbar, hat allerdings Schwächen in Sachen Grafik. Die Sprites sind ziemlich mickrig und besitzen ungewöhnliche Farbgebungen. Oder haben Sie schon mal einen roten Abfangjäger gesehen?





Sonic Boom

Kid kramt in der Sporttasche und findet Fäustlinge seines Vaters. Er streift sie über, und plötzlich findet er sich in der Steinzeit wieder. Als er wilde Tiere erblickt und keulenschwingenden Höhlenmenschen begegnet, wird in ihm nur ein Wunsch geweckt: "Nix wie nach Hause!". Und dabei soll der Spieler helfen. Kid hüpft stets von links nach rechts über den Bildschirm - immer, wenn er den Rand erreicht, gibt es einen riesigen Ruck, bevor das nächste Bild auf dem Screen erscheint. Um sich der vielen Monster zu erwehren, kauft der kleine Held Extrawaffen. Damit lassen sich alle Bösewichter vom Bildschirm pusten. "Kid Gloves" ist ein amüsantes Hüpf- und Springspielchen. Grafik und Sound sind zwar nur durchschnittlich, dafür bietet das Game aber eine ordentliche Portion Spielspaß.





Kid Gloves

Magic Bytes geizt nicht mit den Reizen von Roboterfrauen. Im Mittelpunkt von "Blue Angel 69" steht aber weniger eine aufreizende Blechbeschau, sondern vielmehr das simple und zudem noch fesselnde Spielprinzip. Es geht darum, nach und nach Steine von einem Spielbrett wegzunehmen und auf diese Weise ein Gitter freizulegen, hinter dem ein Roboterweibchen posiert. Das Gitter öffnet sich, wenn alle Steine vom Brett verschwunden sind. Erst dann kann der Spieler das Roboterweibchen in seiner vollen Schönheit bewundern. Ein Spieler zieht immer nur die Steinchen aus waagerechten Reihen, während für die anderen die senkrechten bestimmt sind. Es geht nun darum, den Gegner durch geschickte Planung in eine ungünstige Auswahlposition zu bringen, so daß er in der aktuellen Spalte oder Reihe nur Steine negativer oder niedriger Punktzahl aufnehmen kann. Auch wenn die Grafiken von "Blue Angel 69" etwas hausbacken wirken und genauso erotisch wie eine verfaulte Kartoffel sind, intelligenten Spielspaß bietet "Blue Angel 69" allemal.





Blue Angel 69

Castle Master

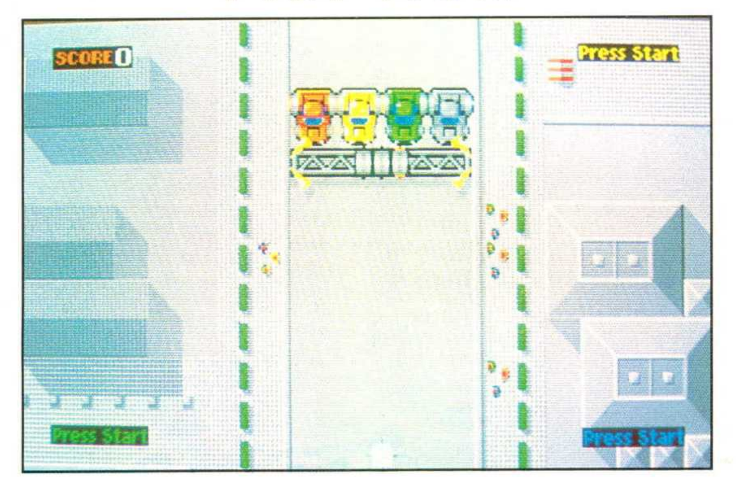


Der berüchtigte Herrscher von Castle Eternity hält eine Person gefangen. Dies wäre nichts Besonderes, wenn es sich nicht um den Zwillingsbruder des Spielers handeln würde. Der Auftrag ist klar: die Behausung erforschen und Familienzusammenführung betreiben. Da der Bösewicht dabei nicht tatenlos zusehen wird, erweist sich die Befreiungsaktion als schwierig. Zu Beginn steht man vor der Burg und sucht nach einem Weg, um ins Innere zu gelangen. Auf dem Weg müssen zehn Schlüssel gefunden werden, ohne deren Besitz das Spiel nicht lösbar ist. Von Zeit zu Zeit greifen Geister an. Dann sind Reaktion und genaues Zielen gefragt, um sie wegzuputzen. Um sich nicht zu verlaufen, ist es empfehlenswert, einen Plan zu zeichnen und Räume und Gegenstände zu vermerken. Für das Abschießen der Geister und das Finden der Schlüssel erhält man einen Punktebonus. Der Joystick erweist sich als unbrauchbar, da nur ein Teil der Funktionen damit ausgelöst werden kann. Nach kurzer Eingewöhnungszeit kann man per Tastatur flott durch die Burg ziehen. Der Bildschirm ist logisch und übersichtlich aufgebaut. Der größte Teil wird mit einer dreidimensionalen Darstellung ausgefüllt. Auf Schnickschnack hat man verzichtet, dafür umso mehr Augenmerk auf die relevanten Gegenstände gelegt. Der 3D-Effekt ist hervorragend gelungen. Links neben dem Hauptbildschirm ist ein Turm mit drei Öffnungen zu erkennen. Durch Anklicken wird hier die Art der Fortbewegung geändert. Im Freien und auf überschaubaren Flächen kann man auf "Rennen" umschalten; wenn es ganz eng wird, muß man sich auch mal kriechend durch den Schlamm wälzen. Unterhalb des Spielfelds finden sich einige Angaben wie eine Kraftanzeige, ein Schlüsselbrett mit den bereits gefundenen Schlüsseln und einige Icons zur besseren Orientierung. Ein Fenster in der Mitte gibt Informationen darüber, wo man sich gerade befindet . Neben der sehr guten Grafik tragen auch die Soundeffekte zur Atmosphäre dieses Adventures bei. Die Rätsel sind teilweise schwer, so daß man einige Zeit beschäftigt ist. Leider werden die "äußeren Umstände" (Anleitung, Übersetzung) "Castle Master" nicht gerecht und mindern den ansonsten guten Gesamteindruck.

FK/CBO

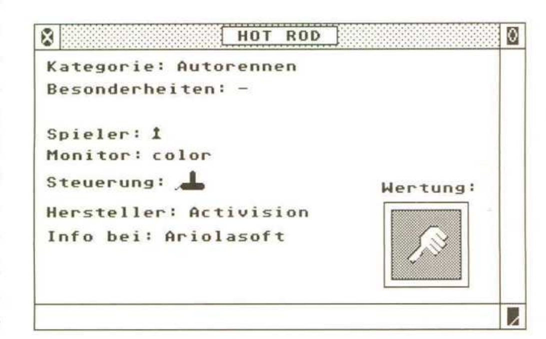

CASTLE MASTER
Kategorie: Actionadventure
Besonderheiten: -
Spieler: 1
Monitor: color
Steuerung:
Hersteller: Domark
Info bei: Bomico
Wertung:

Hot Rod



Bis zu vier Spieler gehen an den Start. Fünfzehn verschiedene Rennstrecken gilt es zu meistern. Die Steuerung sieht nur das Notwendigste vor.: rechts/links und Gasgeben. Freunde komplexer Rennsimulationen werden von "Hot Rod" enttäuscht sein. Stattdessen gibt es Action pur, lediglich unterbrochen durch die Möglichkeit, im Ersatzteilladen seinen Flitzer mit neuen Extras auszustatten. Doch auch hier wird nichts angeboten, was man nicht schon von anderen Autorennspielen her kennt: stärkere Motoren, neue Reifen und "Flossen" für bessere Straßenlage. Bezahlt wird mit Geld, das man durch gute Plazierungen erhält. Nach Beendigung jeder Rennstrecke bekommt man eine bestimmte Menge Sprit , die für die nächste Runde ausreichen muß. Das Hauptziel bei "Hot Rod" liegt nicht darin, das Rennen als Erster zu beenden, sondern im Erreichen der Ziellinie. Allein das Ankommen berechtigt schon zum Weitermachen. Allerdings ist es dennoch nicht allzu ratsam, den Fuß vom Gaspedal zu nehmen, da sich das Bildschirm-Scrolling nach dem führenden Wagen richtet. Sobald das eigene Auto aus dem sichtbaren Bildschirmbereich zu verschwinden droht, "fliegt" es wie von Geisterhand bewegt über sämtliche Hindernisse hinweg an die Spitze des Feldes. Die Rennstrecke wird aus der Vogelperspektive dargestellt. Die Auto-Sprites sind klein, aber gut zu erkennen. Die gelegentlich eingestreuten Hindernisse wie Polizeiautos und Bauarbeiter sind nur schwer als solche zu erkennen. Positiv hervorzuheben ist die Geschwindigkeit, mit der das Rennen abläuft. Die Autos bewegen sich in flottem Tempo über den Bildschirm, was einiges an Geschick erfordert. Dennoch hätten sich gerade Profis sicherlich einen höheren Schwierigkeitsgrad gewünscht. Obwohl die Pisten immer nachgeladen werden müssen, sind die Wartezeiten kurz. Leider werden die erzielten Ergebnisse nicht in der Highscore-Liste verewigt, so daß die Motivation rapide fällt, wenn man alle fünfzehn Rennstrecken bewältigt hat. Somit gehört auch "Hot Rod" zu den Spielen, die nach kurzer Zeit in der Diskettenbox verschwinden.

FK/CBO


HOT ROD
Kategorie: Autorennen
Besonderheiten: -
Spieler: 1
Monitor: color
Steuerung:
Hersteller: Activision
Info bei: Ariolasoft
Wertung:



Eingeschränkte Versionen kommerzieller Software nehmen wir gerne in die 2000er Serie auf, wenn sie sich von der Vollversion im wesentlichen nur in der Anzahl der zu bearbeitenden Datensätze, bzw. der mitgelieferten Spielebenen unterscheiden. Bei privater Nutzung werden die "Grenzen" der Demo-Version oft erst nach Monaten erreicht, einem intensiven Test und der möglichen Kaufentscheidung zu Gunsten des Profiprodukts steht nichts im Wege.

Daß PD-Versionen beim Erwerb von Updates oder Upgrades angerechnet werden, ist schließlich ein weiterer Schritt in Richtung Anwenderfreundlichkeit. Hinweise darauf finden Sie bereits in vielen PD- und Sharewareprogrammen.

Mit freundlichen Grüßen,

Die "23"

2131



Adresswriter erstellt und verarbeitet Adressenlisten, Serienbriefe und ähnliche Mischformulare auf einfache Weise. Mit seiner komfortablen GEM-Oberfläche und dem integrierten Texteditor ist der Adresswriter einfach zu bedienen (s/w).

Discothek wird professionellen Ansprüchen bei der Verwaltung von CDs, LPs und MCs gerecht. Eine schnelle und bequeme Suchroutine erleichtert das Auffinden gewünschter Titel (s/w).

Das Textdruckprogramm **Idealist** ist in der neuen Version **2.0** erschienen. Die stark überarbeitete Benutzerführung neue Optionen und verbesserte Druckertreiber sind hinzugekommen. Jetzt sollte wirklich jeder Drucker Idealist verstehen (s/w).



Schorni ist eine Datenverwaltung für Schornsteinfeger (s/w).

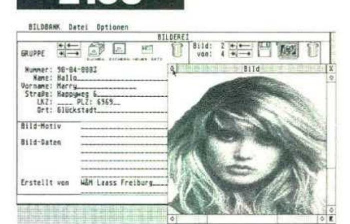
2132

Vier Probebildschirme von **Esprit** stehen in dieser PD-Demo-Version des gleichnamigen Spieles von Application Systems zur Verfügung. Probieren Sie's? (s/w)

2133

Blöcke ist eine dreidimensionale Tetrisvariante. Die Teile fallen nun von oben in einen Schacht hinein und müssen so angeordnet werden, daß der Boden gleichmäßig bedeckt ist (f).

Lunacy ist eine besonders schöne Tetrisvariante für Farbbildschirme (f).



Stoneage von Motelsoft setzt ein geschultes Auge und gutes Gedächtnis voraus. In dieser Shanghai-Variante werden jeweils zwei gleiche Steine vom Brett genommen, wenn der Spieler sie nach oben, unten oder zu einer Seite wegziehen kann (f).

Quartet nennt sich die digitalisierte Sound-Demo von Microdeal.

2134

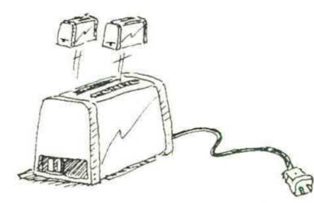
Transmitter GmbH ist ein grafisch und spielerisch hervorragend gemachtes Weltraumhandelsspiel, das den Vergleich mit kommerzieller Software nicht scheuen muß. Der fliegende Händler reist von Planet zu Planet und versucht mit attraktiver Handelsware möglichst hohe Profite zu erzielen (f).

2135

Bildbank dient der Verwaltung von Bilddaten im Degas-, NEO-, ART- und IMG-Format. Bei dem Programm handelt es sich um ein vielseitig verwendbares Informationssystem, bei dem Bilder, Begleittexte und Bedienungshinweise jeweils in separaten Fenstern dargestellt werden. Bildbank läuft auf sämtlichen Monitoren – auch auf Großmonitoren und Big-Screen Erweiterungen – und verfügt über eine Graustufendarstellung für Farbbilder. Sogar an den Einsatz einer Monitor-Switchbox wurde gedacht, damit Farb- und Monochrombilder auf dem dafür vorgesehenen Bildschrim dargestellt werden können.

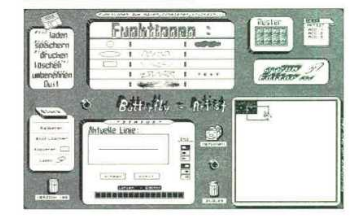
Zeichenplatte stellt einige Sonderfunktionen bereit, die in herkömmlichen Malprogrammen nicht zu finden sind. Mit Kontrastverstärkung, Verschiebefunktion und Vektorschrift werden interessante Bildmanipulationen vorgenommen (s/w).

2136



Clip-Art 13 enthält wieder viele schöne Bilder für DTP- und Textprogramme.

2137



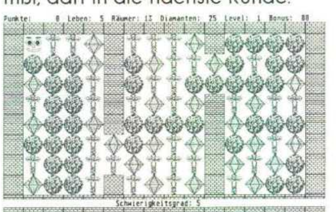
Butterfly-Artist bietet zahlreiche Zeichenfunktionen, die sonst nur in sehr teuren Profiprogrammen zu finden. So bietet es eine Zeichenfläche, die nicht durch den Bildschirm begrenzt ist, eine vielseitige Lassofunktion und die Möglichkeit, zusätzliche Zeichensätze einzusetzen. Das originelle Hauptmenü birgt einige Überraschungen (s/w).

ST-Design ist eines der leistungsfähigsten PD-Zeichenprogramme für den ST. Das variable Hauptmenü enthält viele Sonderfunktionen, wie drehbare Rechtecke, variable Lupe, Linienverlauf und Vektorschrift. Der beiliegende Vektor-Fonteditor läßt Ihrer gestalterischen Ader freien Lauf (s/w).

2138

Hinter **Chicago** verbirgt sich ein Kneipenspielchen der geselligen Art: Jeder der beiden Spieler (Spieler und Computer) nehmen 1-3 Streichhölzchen in die Hand und tippt dann wieviele beide zusammen in Händen haben. Der Gewinner verliert ein Streichhölzchen. Wer zuerst keines mehr besitzt ist Sieger der Runde und bekommt einen Schnaps. Gesamtsieger ist der erste Spieler mit 15 Schnäpsen (s/w).

Der Wurm versucht alle Punkte seines Spielfelds aufzuessen. Daß das Tierchen nach jeder Malzeit ein Stück länger und damit schwieriger zu steuern wird, erfordert viel Geschicklichkeit vom Spieler. Denn nur wer das Spielfeld innerhalb des Zeitlimits leerfrißt, darf in die nächste Runde.



Face's Revenge ist eine Boulder Dash-Variante für Farbe und s/w. Die Spielfigur soll verstreute Diamanten aufsammeln und dabei herabstürzenden Felsbrocken ausweichen. Das knapp bemessene Zeitlimit bringt gepflegte Hektik ins Spiel und wem das nicht genug ist, der kann sich das Leben so richtig schwer machen: der eingebaute Leveleditor schafft Abwechslung.

Kniffel ist eine neue Version des bekannten Spiels. Man braucht sich selbst keine Gedanken zu machen, was die Würfel bedeuten, da das Programm alle Möglichkeiten vorgibt, bei denen Punkte zu machen sind.

Beim Frage- und Antwortspiel **Knowmeier** wählt jeder der 1-3 Spieler die Richtige von drei

Besonderheiten:

f	= Läuft nur in Farbe
s/w	= Läuft nur monochrom
E90	= Läuft auch mit Emula 5.1 von Diskette 2090.
e	= in englischer Sprache
J	= Joystick notwendig
MB	= 1MB RAM erforderlich
S	= Shareware
G	= GFA-Quellcode liegt bei
K	= Kontaktkarte eingebaut

möglichen Antworten aus. Abhängig von der Zahl der korrekten Antworten gibt's Punkte für die Teilnehmer, die sich anschließend noch einem spannenden Extra- und Risikospiel unterziehen. Die Wissensgebiete Hauptstädte, Personen und Computer sind vorbereitet (s/w).

Super Grips spielt nur der, der ihn besitzt. Eine Datei mit kleveren Fragen liegt diesem anspruchsvollen Denkspiel bei. Wer's noch härter will, nutzt den internen Editor für neue Herausforderungen.

2139

Quicklearn ist ein variabler Vokabeltrainer mit zahlreichen Abfragedateien für Englisch, Französisch, Morsealphabet, Fremdwörter, Kalorien, Währungen usw. Zusätzliche Vokabellisten können vom Anwender auf einfache Weise selbst erstellt werden, wobei die Abfragerichtung – Deutsch – Englisch oder Englisch-Deutsch – den Anforderungen gemäß wählbar ist (s/w).

Mit **Video** können Sie Ihre Sammlung bequem katalogisieren. Eine große Eingabemaske enthält dann alle wichtigen Daten zu den Filmen. Auf der Diskette befindet sich jeweils eine spezielle Version für VHS- bzw. Beta-Videosysteme.

2140

Alibi erzeugt eine kurzes Starterprogramm, welches das eigentliche Hauptprogramm aufruft. Dabei kann das Hauptprogramm irgendwo auf der Festplatte sein, Alibi findet es bestimmt.

Cache erzeugt einen Cachespeicher, der oft benötige Sektoren im RAM hält. So verkürzt sich die Zugriffszeit beträchtlich.

Cassette 3.2 bietet die Möglichkeit, Etiketten für Musikkassetten auf einfache Weise zu erzeugen und auszudrucken (s/w).

Filedup ist ein komfortables Filecopyprogramm. Gewünschte Dateien werden durch Anklicken selektiert und dann gemeinsam eingelesen. Einmal im Speicher, können Sie beliebig oft geschrieben werden.

Auf der Diskette befinden sich zwei **HD-Wait**-Programme. Eines ist als Auto-Lader konzipiert und bietet zudem die Möglichkeit des Bootens von Disk. Das Andere erzeugt einen Bootsektor, von dem aus die Verzögerung mittels Programm eingeleitet wird.

Minilist ist ein hilfreiches ACC, mit dem beliebige Texte vom Desktop aus geladen und dann bequem angesehen werden können. Selbst rückwärts blättern ist möglich.

Planet holt den Himmel auf Ihren Monitor. Das Astronomieprogramm erlaubt den Blick auf den Nachthimmel von jeder Position aus. Zusätzliche Himmelskörper (z.B. Kometen) können vom Anwender eingegeben werden, obwohl auch hier schon die wichtigsten eingebaut sind. Sterne und Planeten können beliebig an- und ausgeknipst werden (s/w).

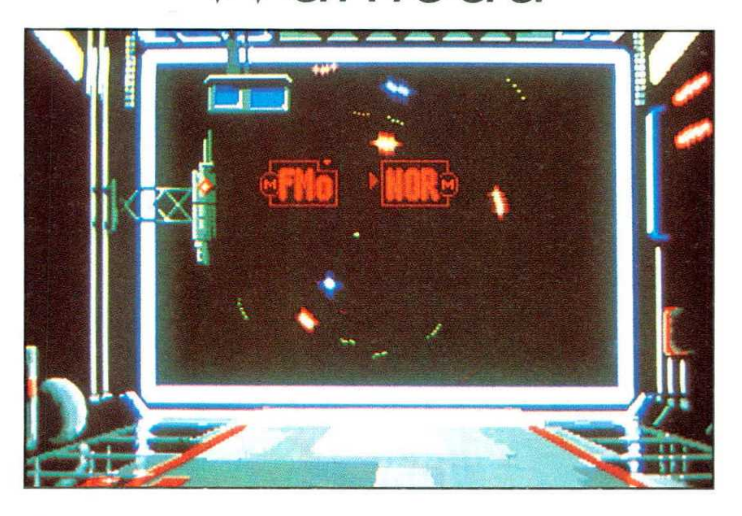


Ramdisk ist eine neue Ramdisk, die garantiert auch unter dem neuen TOS 1.4 funktioniert. Sie ist resetfest, konfigurierbar und lädt ihre Daten beim Booten.

Startup erlaubt es beim Booten aus den AUTO-Programmen und ACC auf der Diskette die benötigten herauszusuchen.

Das schnelle Harddisk-Backup-Programm **Turtle** liegt nun in der neuesten Version **3.0** vor. Zahlreiche nützlichen Hilfsprogramme sind im Lieferumfang enthalten.

Warhead



Mitte des 21. Jahrhunderts steht die Erde mal wieder vor einer existentiellen Bedrohung. Eine insektenähnliche Rasse aus einem benachbarten Sonnensystem greift an. Das Ende steht bevor, doch es gibt noch eine kleine Chance, dem Würgegriff der unheimlichen Fremden zu entgehen. Für das Raumschiff FOE.57 wird ein Kommandant gesucht, der Mut und Geschick hat, die Entscheidungsschlacht zu führen. Nach einer imposanten Eröffnungssequenz befindet man sich mitten im Geschehen. Das Raumschiff wird zwar vorwiegend mit der Maus gesteuert, doch viele wichtige Funktionen sind nur per Tastatur erreichbar. Grundsätzlich lassen sich die Tasten in drei Gruppen zusammenfassen. Über die Zifferntasten des Hauptfeldes wird der Autopilot bedient, der nach Wunsch einen Teil der Steuerung übernimmt. Die Funktionstasten aktivieren die Waffensysteme, während mit einigen Buchstabentasten verschiedene Informationsseiten aufgerufen werden. "Warhead" besteht aus insgesamt 39 Missionen, die alle der Reihe nach gelöst werden müssen. Der Schwierigkeitsgrad steigt langsam an. Während die ersten Missionen nur dazu dienen, sich mit dem Raumschiff vertraut zu machen, wird es spätestens ab der zehnten Mission ganz schön hektisch. Gute Reaktion und sicheres Zielvermögen sind Voraussetzung, um erfolgreich zu bestehen. Nach Beendigung jeder Aufgabe besteht die Möglichkeit, den Spielstand auf Diskette abzuspeichern. Das Geschehen wird hauptsächlich aus der Sicht des Piloten dargestellt. Lediglich beim Aufruf von Informationsseiten (wie Sonnensystemkarte oder Taktikseite) ändert sich der Bildschirm und zeigt die erforderlichen Daten an. Die Weltraumgrafik ist detailliert gezeichnet (alle wichtigen Punkte lassen sich problemlos identifizieren) und läßt auch in bezug auf Geschwindigkeit keine Wünsche offen. Die Maussteuerung ist sehr präzise, wenn auch gerade für Anfänger in Verbindung mit der Tastatur nicht ganz ohne Probleme. Mit einiger Übung kommt man jedoch gut damit zurecht. Für ganz unerfahrene (oder untalentierte?) Flieger wird die erste Mission im Handbuch schrittweise erklärt.

FK/CBO

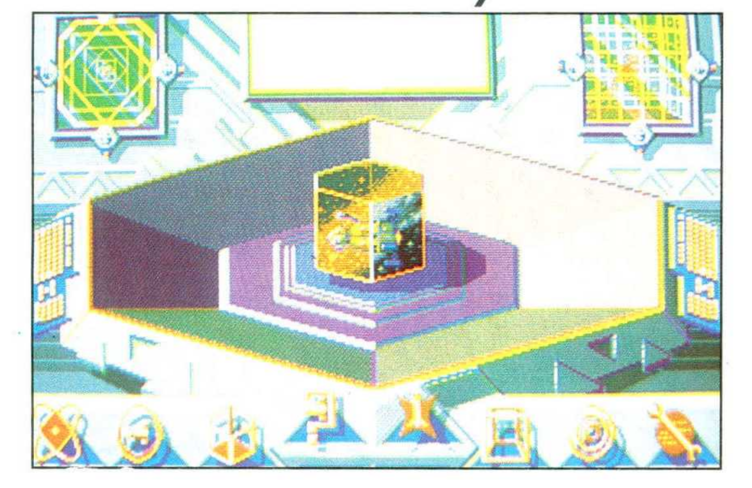

WARHEAD

Kategorie: Weltraumabenteuer
Besonderheiten: -

Spieler: 1
Monitor: color
Steuerung:
Hersteller: Activision
Info bei: Ariolasoft

Wertung:

Gravity



Der Weltraum im Jahre 2320. Das Reisen zu anderen Sternensystemen ist längst selbstverständlich geworden. Doch plötzlich tauchen die OUTIES auf und versuchen, sämtliche Energie der Galaxis an sich zu reißen. Aufgabe des Spielers ist es, sich dem Kampf gegen die Eindringlinge zu stellen. Die einzige Möglichkeit, den Sieg zu erringen, liegt in der Zerstörung des schwarzen Lochs im Sonnensystem der OUTIES. Ein Spiralnebel muß erforscht und die Aktionen der OUTIES durch Kolonisierung neuer Planeten beobachtet werden. Der Spieler ist Commander des Raumgleiters HAWKING. Ein würfelförmiger Holotank ermöglicht eine effektvolle, dreidimensionale Darstellung der Galaxis.Die zur Verbesserung der Schiffsausrüstung erforderlichen Mittel werden durch das Zerstören von feindlichen Schiffen aufgetrieben. Die DRONES (kleine Raumschiffe) können mittels einer eigenen Sprache programmiert werden und lassen sich zu willigen Kampfmaschinen umfunktionieren. Das Geschehen spielt sich bei Gravity ausschließlich in den eigenen vier Wänden des Scoutschiffes ab. Wer gerne mit einem modernen Raumschiff auf Außerirdische Jagd machen möchte, sollte sich auf jeden Fall nach einem anderen Programm umsehen. Alle am Spielgeschehen beteiligten Objekte werden dreidimensional ins Innere der Raumfähre projiziert. Diese Sequenz ist zwar grafisch sehr gut gelungen , doch geht dadurch leider ein Teil des Weltraum-Feelings verloren. Etwa zwei Drittel des Screens sind für die Darstellung der Objekte reserviert. Am unteren Rand befinden sich einige Icons zur Auswahl der Optionen. Für ein schnelles Spiel "zwischendurch" ist Gravity sicherlich nicht geeignet. Wer sich allerdings gerne komplexen Aufgaben stellt und die Geduld für mehrere Tage oder gar Wochen aufbringt, ist mit diesem Programm recht gut ausgerüstet. Voraussetzung ist allerdings, daß man sich die Zeit nimmt, die etwa 80seitige deutsche Anleitung aufmerksam durchzulesen und sich mit der relativ komplizierten Mausbedienung vertraut zu machen.

CBO/FK


GRAVITY

Kategorie: Handelssimulation
Besonderheiten: -

Spieler: 1
Monitor: color
Steuerung:
Hersteller: Mirrorsoft
Info bei: Ariolasoft

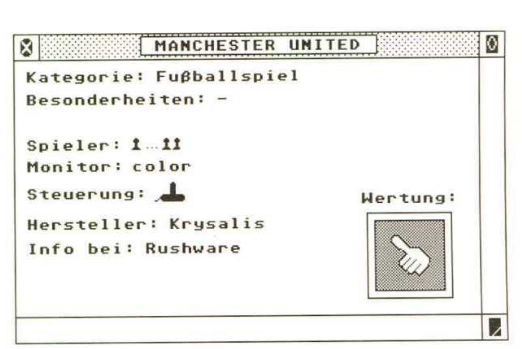
Wertung:

Manchester United

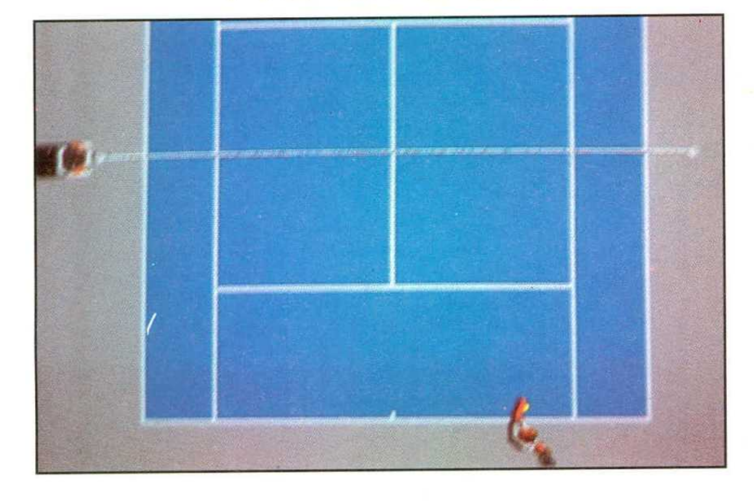


Zwei Spielgenres sind hier gelungen miteinander verknüpft worden : Football Manager und ein aktionsgeladenes Fußballspiel. Beginnen wir mit dem Management. Es ist möglich, in verschiedenen Tafeln Informationen über das gesamte Team aufzurufen. Jeder Spieler wird vorgestellt. Für einen Sieg ist es erforderlich, immer die ausdauerndsten und schnellsten Kicker aufzustellen. Die Spieler können gekauft werden. Angebot und Nachfrage regeln das Geschäft. Ums Geld brauchen Sie sich keine Gedanken zu machen. Kapital ist unbegrenzt vorhanden. Wer keine Lust hat, mit Zahlen zu jonglieren, und sich viel lieber auf dem grünen Rasen wälzt, kann den Managerteil einfach abstellen. Ideal für actionfreudige Managermuffel! Wer gerne mit dem Joystick kickt, kann ein Spiel bis auf 90 Minuten ausdehnen. Gespielt wird gegen Computer oder Mitspieler. Wer "Manchester United" zu zweit spielt, wird feststellen, daß kaum Tore fallen - die computergesteuerten Torhüter vollführen eine Glanzparade nach der anderen. Ansonsten macht "Manchester United" großen Spaß. Die Steuerung ist einwandfrei. Ohne Probleme lassen sich präzise Eckstöße kicken. Fällt ein Tor, kann man es später noch einmal auf einer Videowand bewundern. Grafik und Sound sind ebenfalls zufriedenstellend. Nicht nur, daß man die Spieler während ihres Treibens auf dem Rasen aus der Tribünenperspektive beobachtet, das Programm kann auch mit gröllenden Zuschauern und animierten Linienrichtern aufwarten. Beim horizontal scrollenden Spielfeld gibt es vor den Toren sogar zertretenen Rasen. "Manchester United" glänzt durch Liebe zum Detail, gute Spielbarkeit und gelungene Präsentation. Trotzdem würde ich Fußballfans nicht unbedingt empfehlen, sofort in den nächsten Laden zu eilen und Manchester United zu kaufen. Andere Software-Häuser wie beispielsweise Ocean, U.S Gold, Codemasters, Linel, Virgin und Microprose bringen in der nächsten Zeit ebenfalls neue Fußballspiele auf den Markt.

CBO

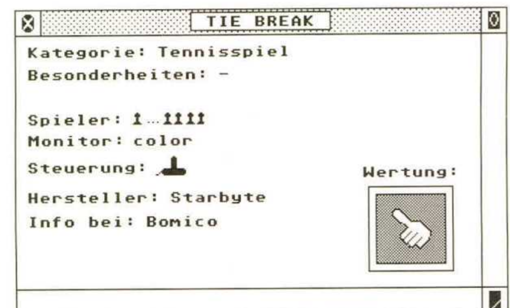


Tie-Break



Mancher Relax-Leser erinnert sich womöglich noch an"Passing Shot", den totalen Tennisflop mit Ballwechseln aus der Vogelperspektive. "Starbyte" hat das Spielprinzip dieser Automatenumsetzung wieder aufgegriffen und ein duftes Game daraus gemacht. Zur Einstimmung geben Zuschauer digitalisierte Geräusche von sich, und der Schiedsrichter schreit "Quiet please!". Wer sich für knapp 10 DM einen Vier-Spieler-Adapter zulegt, kann an seinen ST vier Joysticks anschließen. Dann können sich vier Spieler im Doppel die Bälle um die Ohren hauen. Natürlich lassen sich Doppel auch ohne Partner austragen. Wer keine Mitspieler findet, kann die anderen Tennisspieler vom Computer steuern lassen. Bevor man sich anschickt, bei großen Turnieren wie Wimbledon, den French oder U.S Open teilzunehmen, sind einige Trainingsspiele zu empfehlen. Der Tennisplatz nutzt den gesamten Bildschirm aus. Grafisch haut die Darstellung zwar keinen vom Hocker, ist dafür aber ziemlich zweckmäßig. Ein waagerechter Strich stellt das Netz dar, am linken Rand hockt der Schiedsrichter auf seinem Türmchen. Seine Entscheidungen gibt er in digitalisierter Sprachausgabe von sich. Wenn auch die Darstellung des Platzes ziemlich spartanisch ausfällt, so sind wenigstens die Sprites ganz ordentlich gelungen. Korrekte Schatteneffekte und die perfekten Animationen verzücken das Auge. Mit dem Joystick schlägt man Top Spin, Slice und Schmetterbälle. Glücklicherweise braucht man sich beim Spiel nur auf die Schläge zu konzentrieren, die Laufarbeit übernimmt freundlicherweise der Computer. Die Steuerung der Schläge ist sehr vielseitig und nuancenreich, dadurch leider aber auch sehr gewöhnungsbedürftig. Man kloppt in "Tie-Break" nicht nur auf den Ball, sondern führt mit dem Joystick auch die Schlagbewegung durch. Wer einige Spiele hinter sich hat, wird mühelos mit der Steuerung fertig und hat zudem noch realistischen Spielspaß. Besonders mit dem Vier-Spieler-Adapter toben heiße Matches über den Screen. Da ist Spielspaß garantiert!

CBO

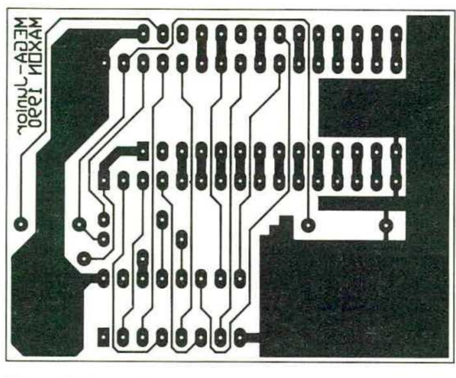


ST EXTRA Die zweite

Da uns recht viele Anfragen zum Sonderheft 3 erreicht haben, wenden wir uns mit dem folgenden Nachschlag an alle Leser des ST EXTRA:

- Leider sind im ST-Computer Hardware-Sonderheft einige Layouts zu den veröffentlichten Schaltungen nicht in der Originalgröße abgedruckt worden. Alle interessierten Bastler finden diese auf den nächsten Seiten 1:1 abgebildet.

- Statt des aktuellen Layouts zum FBAS-Wandler wurde leider eine frühere (Test-) Version abgedruckt. Die richtige Vorlage ist ebenfalls auf den nächsten Seiten zu finden.

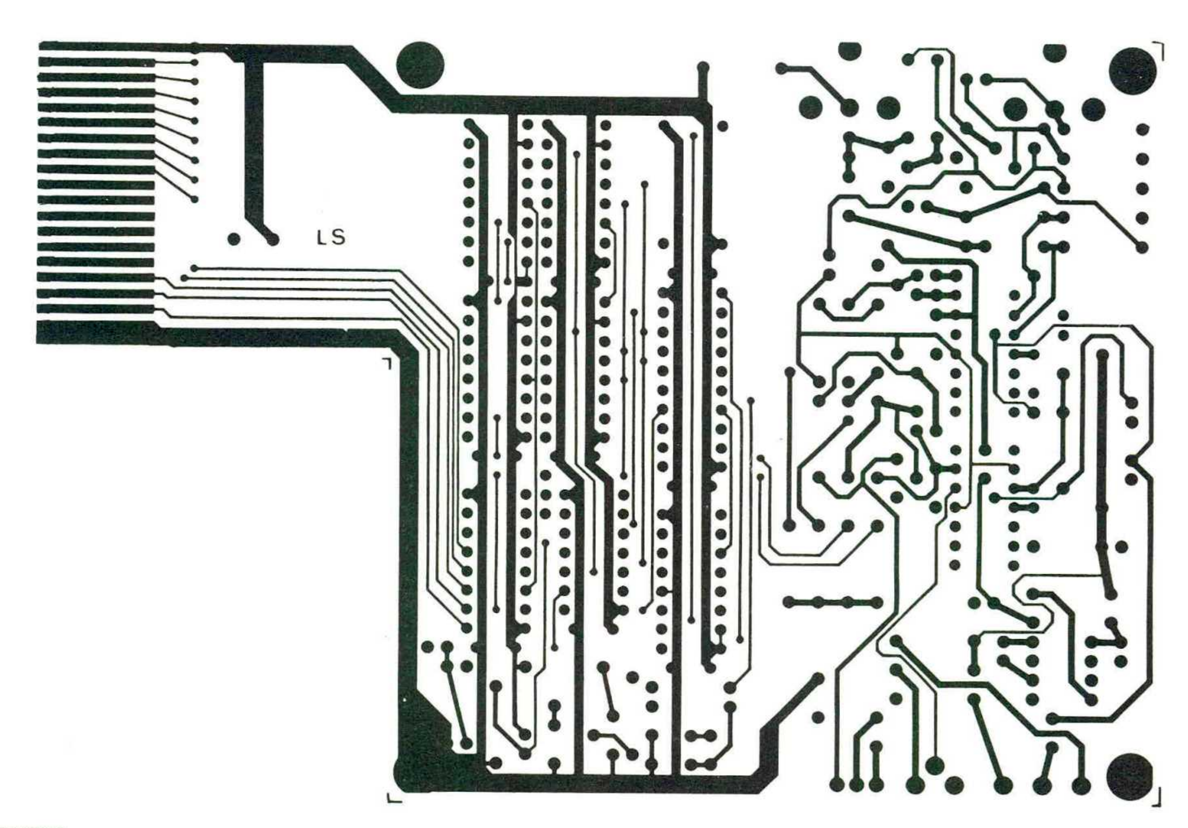


Mega-Adapter Lötseite

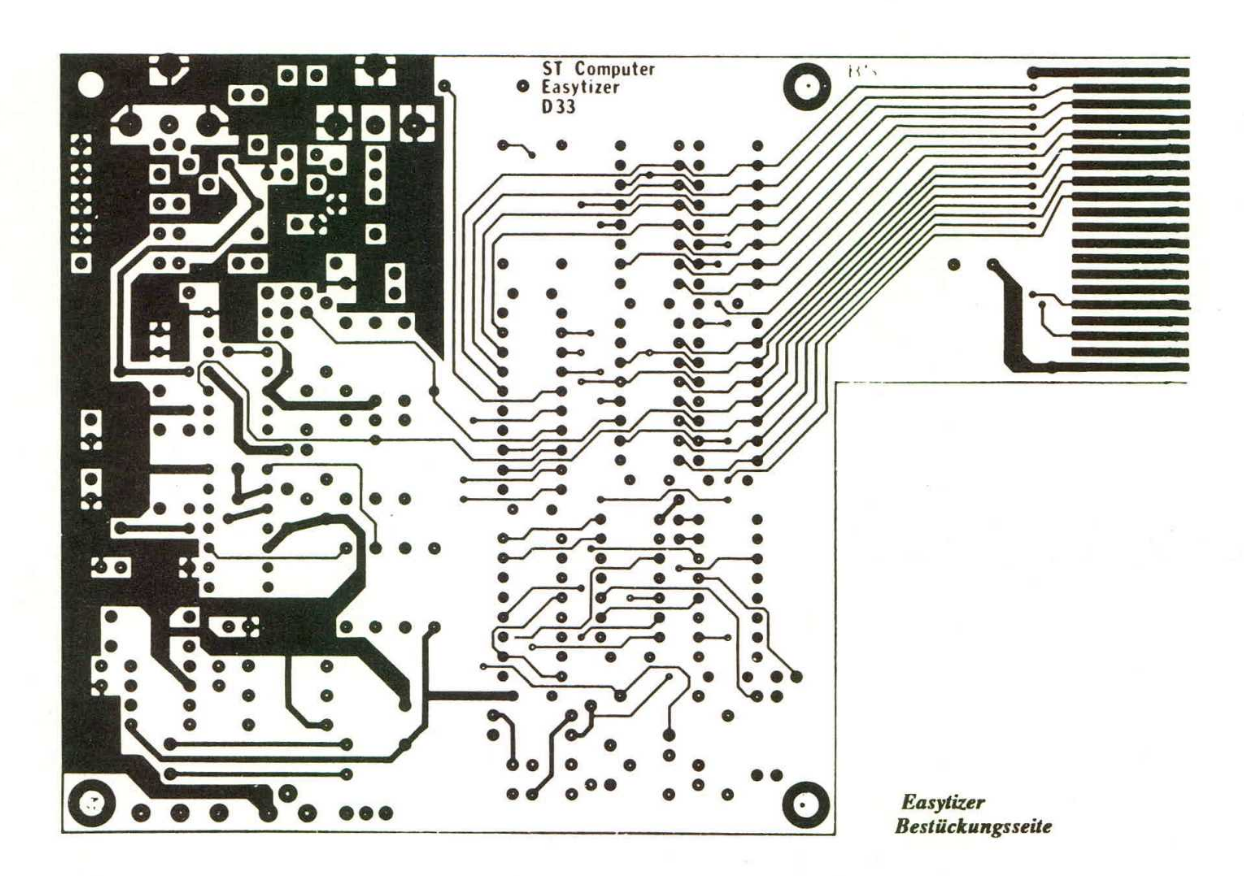
- Auf dem Bestückungsaufdruck zum Junior Prommer fehlt die Zehnerdiode (D5). Sie muß direkt an die Pins 1 und 6 des IC7 angelötet werden (siehe Schaltbild).

- Die Cache-RAMs für die Mach16 scheinen nicht weit verbreitet zu sein. Deshalb hier die Adresse eines Herstellers, bei dem sie bezogen werden können: SCANTEC, Berngstr. 10, 8033 Planeck.

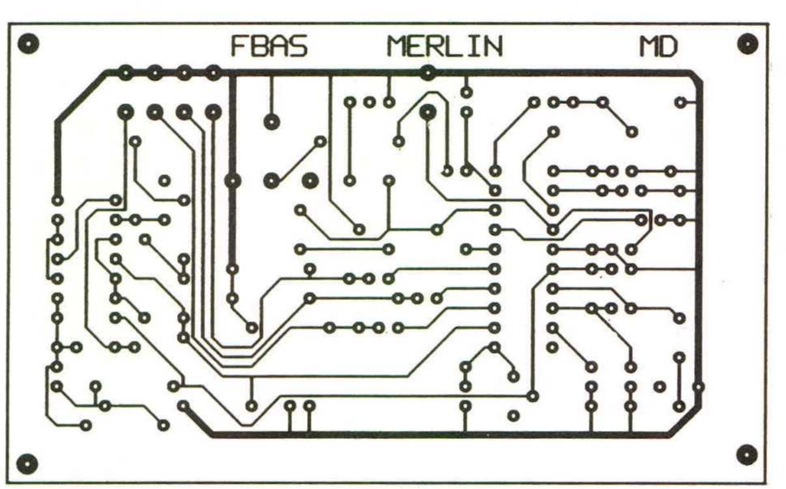
tw



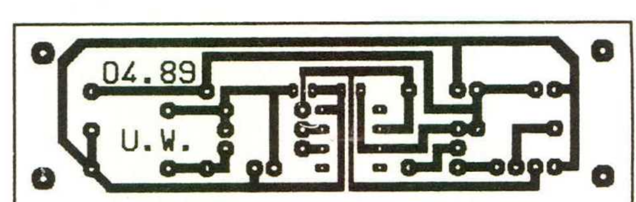
Easytizer Lötseite



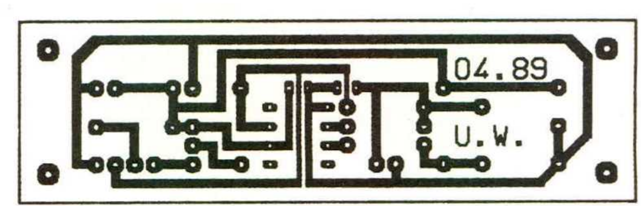
Easytizer Bestückungsseite



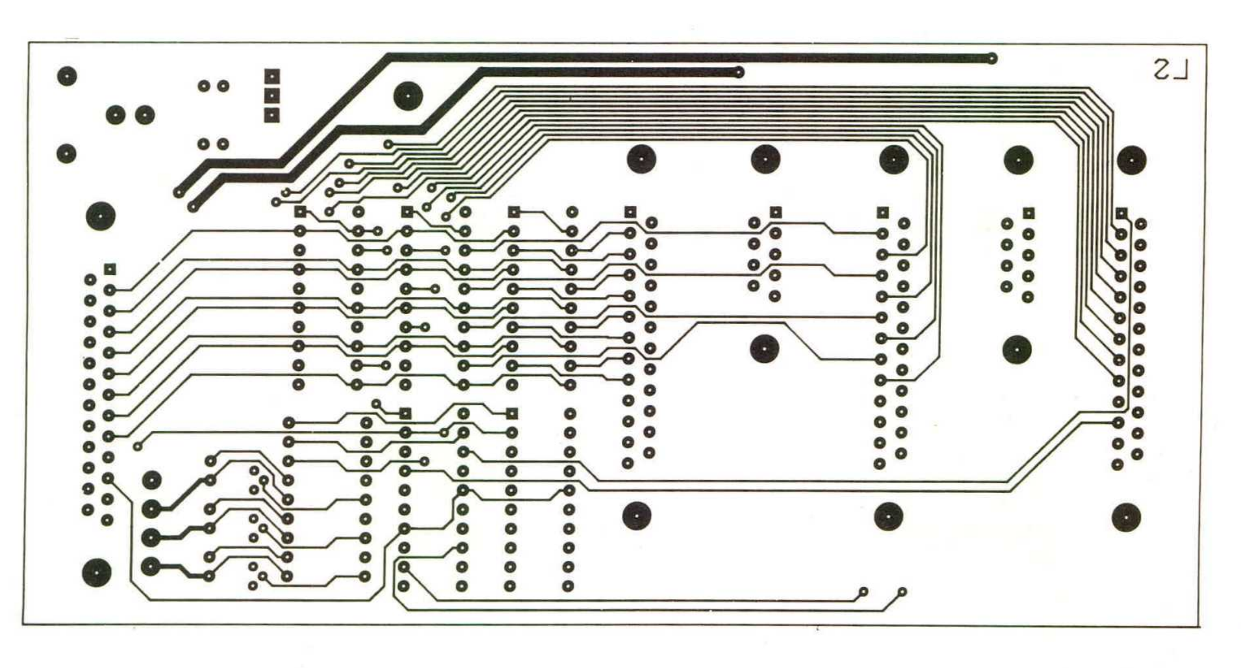
FBAS-Wandler Lötseite



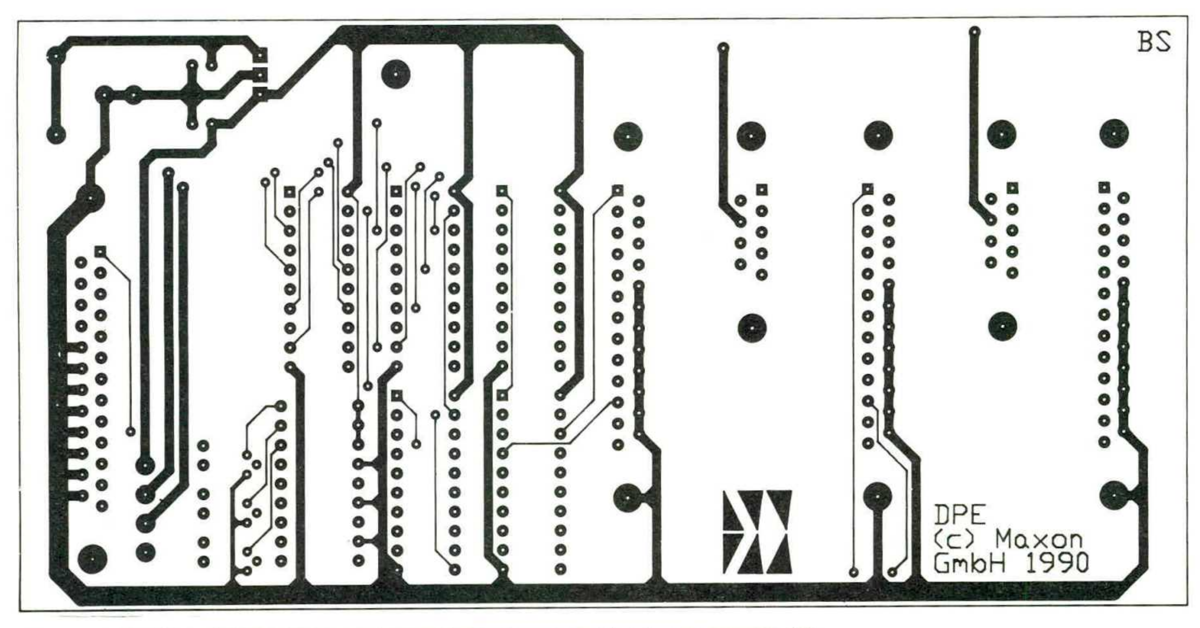
Akkupufferung Lötseite



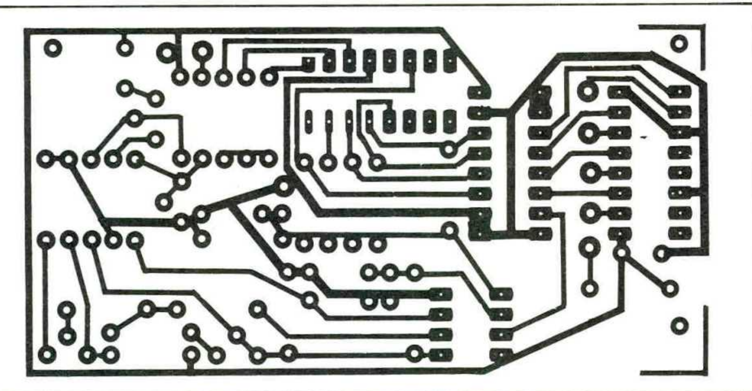
Akkupufferung Bestückungsseite



DPE Lötseite



DPE
Bestückungsseite



Scanner Lötseite

GAL-Prommer

Junior Prommer





MGP - Einfach, schnell und praktisch

Der **MAXON GAL- Prommer 16/20** *(Bestell-Nr.: 890900)* bietet Ihnen alle Möglichkeiten, die bekannten GAL-Typen 16V8 und 20V8 mühelos zu programmieren. Nutzen Sie diese modernen Bausteine, um Ihre Schaltungen schnell, preiswert und flexibel aufzubauen. Mit dem MGP 16/20 kein Problem, denn im ausführlichen Bedienungshandbuch wird auch erklärt, wie herkömmliche Logik (NOR-, NAND-, NOT-, ...-Gatter) in ein GAL programmiert wird und wie diese Bausteine arbeiten.

Der MGP 16/20 läuft an jedem ATARI ST und wird an die Druckerschnittstelle angeschlossen. Die benötigte Betriebsspannung wird am Joystick-Port abgenommen. Die menügesteuerte Software ermöglicht ein bequemes und sicheres Arbeiten mit den GALs. Im eingebauten Editor läßt sich die JEDEC-Datei (so heißt die Datei, die in ein GAL programmiert wird) leicht erstellen oder ändern.

Neben der enormen Platz- und Stromersparnis bleibt eine einmal aufgebaute Schaltung flexibel, denn ein GAL ist in wenigen Sekunden umprogrammiert und kann somit neuen Anforderungen angepaßt werden. Typische Einsatzgebiete eines GALs sind:

1. **Adreßdekoder**
2. **Zustandsautomaten**
3. **logische Gatter**
4. **PAL-Simulation**

Easytizer

Der Videodigitizer ohne Geheimnisse

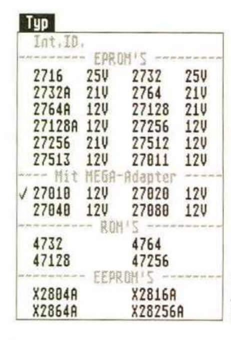
Mit dem **Easytizer** *(Bestell-Nr.: 871000)* können Sie beliebige Videosignale von einer Schwarzweiß- oder Farb-Kamera, Videorecorder oder direkt vom Fernsehgerät (mit Composite Video Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen.
Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden 12,5 Bilder in der Sekunde wiedergegeben. Im hochaufgelösten Modus werden 640x400 Bildpunkte in Schwarzweiß wiedergegeben.

Besondere Merkmale des Easytizers:
•Software vollständig in Assembler • Von STAD und Sympatic Paint ansteuerbar • Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format vergrößert und verkleinert werden • Spiegeln eines Bildes in horizontaler und vertikaler Ebene • Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder) • Eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible • Wahlweise automatische oder manuelle Helligkeitseinstellung • Schnappschuß

Lieferumfang:
1) **Fertiggerät:** komplett aufgebaut und geprüft, inklusive Diskette mit der Easytizer-Software und Bedienungsanleitung
2) **Teilsatz:** Doppelseitiger, elektronisch geprüfter Platine mit Lötstoplack und Bestückungskungsaufdruck sowie vergoldeten Anschlußkontakten,fertig programmiertes GAL 16V8, Quarzoszillatormodul 32 MHz, Diskette und Bedienungsanleitung.

Klein, kompakt und leistungsstark

Der **Junior Prommer** *(Bestell-Nr.: 880310)* programmiert alle gängigen EPROM-Typen (und deren CMOS-Typen), angefangen vom 2716 (2 kByte) bis zum modernen 27011 (1 MBit). Mit dem optionalen **Mega-Modul**-Adaptersockel *(Bestell-Nr.: 880313)* lassen sich sogar die neuesten 32-pol. EPROM-Typen 27010, 27020, 27040 und 27080 brennen. Aber nicht nur EPROMs, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren. Selbstverständlich läßt sich ein 16-Bit Word in ein High- und Low-Byte zerlegen. 5 Programmieralgorithmen sorgen bei jedem EPROM-Typ für hohe Datensicherheit. Im eingebauten Hex/ASCII-Monitor läßt sich der EPROM-Inhalt blitzschnell durchsuchen oder auch ändern.



Typ

Int.ID.

EPROM'S			
2716	25V	2732	25V
2732A	21V	2764	21V
2764A	12V	27128	21V
27128A	12V	27256	12V
27256	21V	27512	12V
27513	12V	27011	12V
Mit MEGA-Adapter			
✓ 27010	12V	27020	12V
27040	12V	27080	12V
ROM'S			
4732		4764	
47128		47256	
EEPROM'S			
X2804A		X2816A	
X2864A		X28256A	



Übersicht der mit dem JuniorPrommer programmierbaren Speichertypen

Haben auch Sie eine Hardware-Schaltung für den ATARI ST entwickelt und suchen noch einen zuverlässigen Vertrieb? Dann schicken Sie uns doch einfach mal ein paar Worte unter dem Stichwort *Hardware-Projekt*.

Vertrieb in der Schweiz:
DTZ DataTrade AG Landstrasse 1 CH-5415 Rieden-Baden Tel.: 056/821880 Fax.: 056/821884

Vertrieb in Österreich:
Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling
Tel.: 06244/7081-17 Fax.: 06244/7188-3

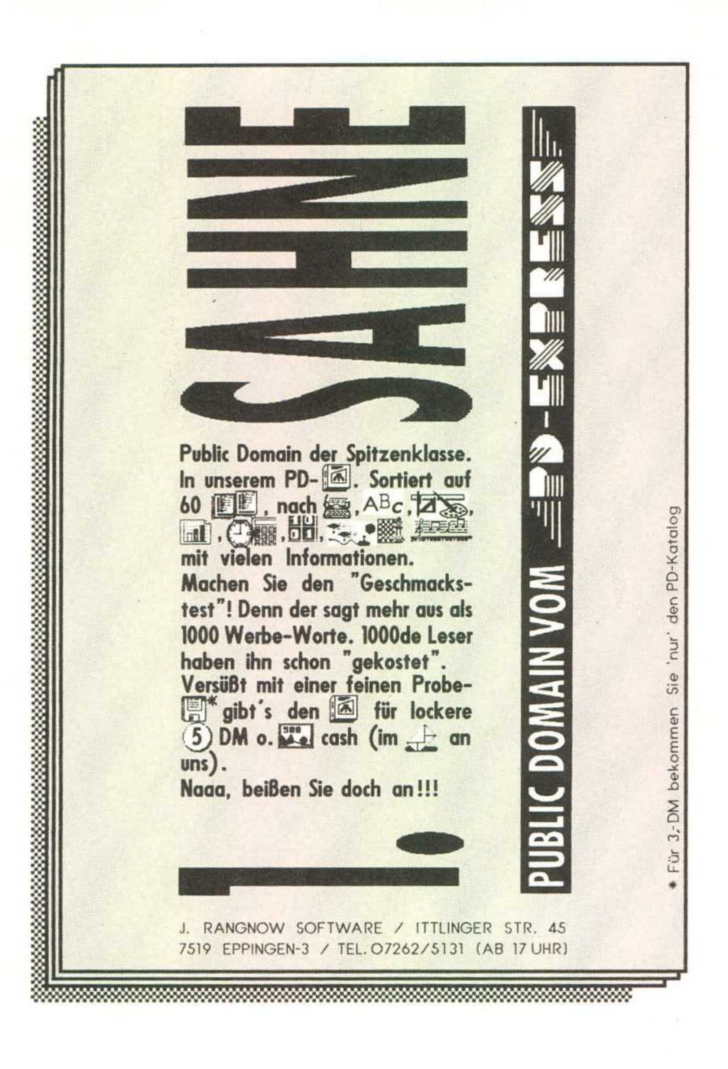
Vertrieb in Frankreich:
AROBACE 2Rue Piemontesis F-75018 Paris Tel.: 1/42235044 Fax 1/42545631

MAXON Computer GmbH
Schwalbacherstraße 52 • 6236 Eschborn
Tel.: 06196/481811 • Fax: 06196/41885

• MGP-GAL-Programmiergerät	DM	195,00
• MGP-Platine, -Software und -Gehäuse	DM	95,00
• Junior Prommer ST (Fertiggerät wie beschrieben)	DM	229,00
• Adaptersockel Mega Modul f. Junior Prommer	DM	99,00
• Junior Prommer Leerplatine und Software (o. Bauteile)	DM	59,00
• Junior Prommer Leergehäuse (gebohrt und bedruckt)	DM	39,90
• ROM-Karte 128 kByte bietet maximal 4 EPROMs Platz (fertigbestückt o. EPROMs)	DM	58,00
• Easytizer (Fertiggerät)	DM	289,00
• Easytizer (Teilsatz wie oben beschrieben)	DM	129,00

Versandkosten:
Inland DM 7,50
Ausland DM 10,00

Auslandsbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.





G R A T I S V E R L O S U N G

Take-off, hochwertige DTP-Grafiken für Publishing-Partner und Calamus®.

Type-Collection Fonts, Vektorfonts für das DTP-Programm Calamus®.

Public-Domain, einzeln auswählbare Programme, je KByte nur 1,5 Pfennige.

Infos anfordern!



ST Profi-Partner
Mönkhofer Weg 126
2400 Lübeck

☎ 0451-505367
FAX 0451-505531

Calamus ist eingetragenes Warenzeichen der Firma DMC.

Besuchen Sie uns auf der Atari-Messe

1st Card

Volltextdatenbank - Hypertext - Grafik
Programmshell - Expertensystemshell
für Ihren ATARI ST



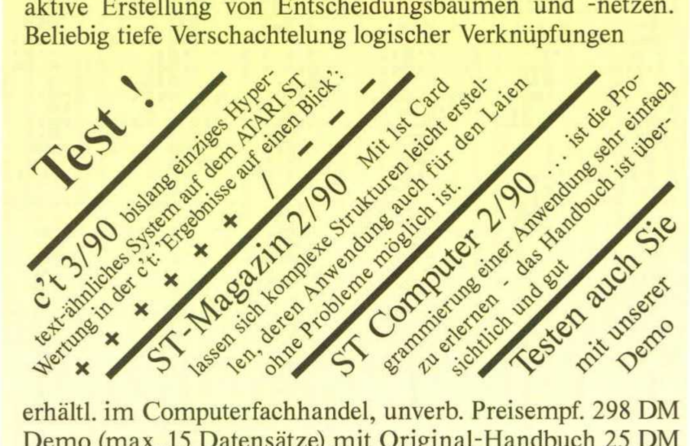
Volltext-Datenbank: freie Positionierung ohne Feld- oder Maskenbegrenzung - Suche nach beliebigen Worten und -fragmenten in 10tel Sek. (auch log. Verknüpfg.)

Hypertext: beliebige Verbindung von Datensätzen durch Buttons per Maus - Suche mit Mausklick auf Worte

Grafik: freie Plazierung von Grafik auf Datensätzen, Import von IMG-Dateien, SNAPSHOT.ACC enthalten

Programmshell: TOS-, TTP-, PRG-Programme können direkt aus Datensätzen mit bestimmten Aufträgen gestartet werden (z.B. Recherche in Online-Datenbanken)

Expertensystemshell: mausgesteuerte interaktive Erstellung von Entscheidungsbäumen und -netzen. Beliebig tiefe Verschachtelung logischer Verknüpfungen



erhältl. im Computerfachhandel, unverb. Preisempf. 298 DM
Demo (max. 15 Datensätze) mit Original-Handbuch 25 DM
Demo-Versand: Bei V-Scheck frei Haus, bei NN + 6 DM

LogiLex Gerhard Oppenhorst, Eifelstr. 32
D-5300 Bonn 1 Tel.: 0228/658346



MDG 19k2-31

Modem mit Pep

Daß auch die DBP Telekom Modems anbietet, dürfte inzwischen bekannt sein. Daß eben diese Geräte jedoch kostenlos für einen Test zur Verfügung gestellt werden, ist ein Novum, das man lange Zeit nicht für möglich hielt. Wir testeten das MDG 19k2-31 für Sie.

Baudraten über 2400 Baud werden immer aktueller. MNP5-, MNP6-Modems oder High-Speed-Geräte bis 19200 bps - immer schneller werden die Leitungsprozessoren. Auch die DBP Telekom mischt in diesem Markt kräftig mit. Unter dem Namen MDG 19k2-31 vertreiben die Postler ein Modem, das auch unter dem Namen "Logem T-2000" oder "Trailblazer 2000" erhältlich ist. All diese Geräte haben eine Gemeinsamkeit: Sie werden von der Firma kabelmetal in Hannover hergestellt und sind größtenteils baugleich. Das Logem T-2000 ist absolut baugleich zum Postgerät, Trailblazer 2000 nennt sich die amerikanische Ausführung des beliebten Geräts.

Das Modem arbeitet mit 300 (V.21), 1200 (V.22) und 2400 Baud (V.22bis). Darüber hinaus ermöglicht es Übertragungsgeschwindigkeiten bis zu 19200 Baud nach einem speziellen Übertragungsverfahren, genannt "PEP". Der Befehlssatz ist Hayes-kompatibel - das jedenfalls behauptet das Handbuch. Tatsächlich eingebaut ist der verstümmelte Hayes-Befehlssatz, da das Modem sonst keine Zulassung bekommen hätte. So läßt sich beispielsweise das Kommando "ATA" (abheben und Antwortton senden) nicht ohne weiteres geben, es funktioniert nur dann, wenn auch unmittelbar vorher das Modem angerufen wurde. Ebenso läßt sich das Gerät nicht dazu veranlassen, die Leitung abzuheben ("AT H1") - der verzweifelte DFÜler wird nur die Meldung "ERROR" erhalten. Der Lautsprecher läßt sich zwar durch Befehle regulieren ("AT L1" bis "AT L3") und man erhält wundersamerweise keine Fehlermeldung, doch der Lautsprecher reagierte bei unserem Testgerät nicht - wahrscheinlich existiert kein solcher (das Testgerät aufzuschrauben, stellt einen strafbaren Tatbestand dar).

PEP oder Papp?

Die wirklich interessante Seite des Modems, nämlich die Geschwindigkeiten über 2400 Baud, sind nur für sehr spezielle Anwendungen benutzbar. Soll das Modem mit 19200 Baud betrieben werden, muß sich auf der anderen Seite der Leitung zwingend ein baugleiches Gerät befinden - mit anderen Standards wie V.29 oder V.32, die in der DFÜ-Szene eigentlich üblich sind, arbeitet das PEP-Protokoll nicht zusammen. 19200 Baud lassen sich beim MDG 19k2-31 aber nur im PEP-Modus (ein Fehlerkorrekturmodus) betreiben. Da in der High-Speed-DFÜ-Szene jedoch kein PEP-Protokoll unterstützt wird, ist das Gerät für diese Anwendung absolut unbrauchbar. Dadurch wird PEP zum Papp. Möglich Anwendungen wären z.B. bei einem DTP-Center zu finden, das große Datenmengen per DFÜ austauschen will, da es hier weniger auf das Protokoll als auf die Datensicherheit ankommt.

Trotzdem sollen noch einige Worte zum Modem selbst gesagt werden. Wie bei allen Postmodems steht dessen Besitzer hier vor dem Problem, daß Mailboxen, die mit einem Lightspeed 2400 C-Modem betrieben werden, nicht mit 2400 Baud angerufen werden können: Das Postmodem schaltet auf 300 Baud herunter und läßt sich nicht dazu bewegen, auf den 2400-bps-Carrier zu warten (was eigentlich ein Problem des Lightspeed-Modems ist, bei fast allen anderen Geräten aber einwandfrei funktioniert). Ansonsten konnten keine Probleme festgestellt werden. Sowohl mit 300, 1200 und 2400 Baud funktioniert das Gerät zuverlässiger als jedes andere Gerät, das ich bisher gesehen habe.

Die Preise sind nicht akzeptabel und verderben jedwede Freude auf den DFÜ-Spaß. Möchte man das Gerät kaufen, ist man mit DM 5300,- dabei - ein für die Leistungen des Modems im Markt nicht haltbarer Preis. Natürlich läßt sich das Gerät auch mieten, in diesem Fall ist eine monatliche Gebühr von DM 142,04 zu entrichten. Soll das Modem von der DBP Telekom gewartet werden, müssen in beiden Fällen zusätzlich DM 38,- pro Monat addiert werden. Soll das Gerät nur dann repariert werden, wenn es wirklich einmal nicht mehr funktioniert, sind DM 5,- für die Grundgebühr, DM 100,- für die Entstörleistung und DM 65,- für die Wegeleistung zu berappen - alles in allem also DM 170,-, die sich bei dem sehr betriebssiche-

ren Gerät wohl eher rentieren als die monatliche Gebühr. Die Anschlußgebühren betragen einmalig DM 65,-.

Fazit

Wer um eine Zulassung seines Modems nicht herumkommt und 5300,- DM zuviel hat, kann sich bedenkenlos das MDG 19k2-31 anschaffen - sofern er in Kauf nimmt, daß er nur 300, 1200 und 2400 Baud sinnvoll nutzen kann. Tip: In einigen Wochen soll das Logem T-2500 verfügbar sein, das nach V.29 oder V.32 arbeiten soll. Mit diesem Standard läßt es sich gut arbeiten. Wird das Modem also nicht sofort benötigt, sollte man lieber noch die Zeit abwarten, bis das T-2500 verfügbar ist. Es wird auch von der DBP Telekom angeboten werden.

Bezugsquelle:
jedes Fernmeldeamt

MP

Crack the West

Der Hamburger Zigarettenmulti Reemtsma sucht den "Hacker" des Jahres. Unter der Telefonnummer (02159) 81008 kann jeder Modembesitzer anrufen und sein Glück versuchen. In verschiedenen Computer-Zeitschriften wird seit einigen Monaten dafür geworben. Hier eine Beschreibung, die direkt aus dem System entnommen wurde: "Herzlich willkommen in der West-Mailbox! In den folgenden 10 Wochen ist Ihr Hacker-Genius gefragt - alle 14 Tage - vom 01.04.90 an - wird sukzessive eine neue Datei eingerichtet, die es der Reihe nach zu 'knacken' gilt. Champ wird, wer zuerst den Lösungs-Code der 5. Datei hat. Reemtsma veröffentlicht - neben dem Titel "Hacker des Jahres" - den Gewinner namentlich in allen einschlägigen Computerzeitschriften. Nehmen Sie die Herausforderung an - fuellen Sie den Antrag korrekt aus - CRACK THE WEST !!!" Damit steht auch schon eindeutig fest, was zu tun ist: Fünf "Lösungs-Codes" sind zu knacken. Wer gewinnt, wird veröffentlicht. Nach Auskunft von Reemtsma will man den geplagten Hackern die Möglichkeit geben, straffrei zu "hacken". Ob der Zigarettenkonzern ein "Herz für Hacker" hat? Normalerweise haben Hacker etwas anderes zu tun, als in fingierten "Datenbanken" herumzuschnüffeln. Das System selbst ist allerdings ziemlich "naturgetreu" nachgebildet, kann aber nur mit DOS-Erfahrung bearbeitet werden. Directories, Programme, Utilities, alles ist in der "Datenbank" vorhanden. Hier gilt es, zu kombinieren. Die Paßwörter, die zum Weiterkommen benötigt werden, sind in verschiedenen Dateien versteckt. Allerdings sind nicht alle Hinweise in der "Datenbank" selbst: Man wird beispielsweise zu Recherchen in der Musikbranche (ein Komponist wird gesucht) gezwungen, um entsprechende Paßwörter finden zu können. Der Gewinner des ungewöhnlichen Wettbewerbs wird in den einschlägigen Computerzeitschriften ausgelobt - sofern er nichts dagegen hat (was bei einem "echten" Hacker doch eher der Fall sein dürfte). Laut Reemtsma soll der Wettbewerb bei entsprechendem Interesse wiederholt werden. Die bisherigen Stimmen sind jeweils allesamt positiv.

MP

Weide ELEKTRONIK

Ladenlokal: Gustav-Mahler-Straße am Einkaufszentrum
viele Parkplätze direkt am Laden

Weide Elektronik GmbH
Regerstr. 34
D-4010 Hilden
Telefon 0 21 03 / 4 12 26
Telefax 0 21 03 / 3 18 20

ATARI System-fachhändler
Leasing für Komplettsysteme möglich
Atari + PC XT/AT

Schweiz
DTZ Data Trade AG
Landstr. 1
CH-5415 Rieden - Baden
Tel. (056) 82 18 80
Fax (056) 82 18 84

Österreich
Reinhart Temmel GmbH
Markt 109
A-5440 Golling
Tel. (0 62 44) 70 81-0
Telefax 7 18 83

STECHUHR ST

Manipulationsgeschützte Software zur Erfassung von Arbeitszeiten mittels Kennungsdiskette für jeden Mitarbeiter

Nutzen

- ➤ verläßliche Stundenlisten
- ➤ individuelle Lohngestaltung und Lohnabrechnung
- ➤ Kostensenkung durch Zeitersparnis und exaktes Datenmaterial

Leistungsmerkmale

- Sekundengenau erfaßte Ankunfts- und Abgangszeiten
- einfache Erfassung durch Einschieben der für jeden Mitarbeiter spezifischen Kennungs-Diskette
- Entnahme der Protokolldiskette akustisch gesichert
- Datum und Uhrzeit werden automatisch über die interne Uhr des Rechners erfaßt
- exakt berechnete Arbeitsstunden pro Mitarbeiter
- beliebig wählbare Auswertungszeiträume
- Protokoll der täglichen Arbeitszeiten auf Diskette und Drucker (optional)
- automatische Fehlerkorrektur bei vergessener Erfassung
- MS-DOS kompatibles Diskettenformat
- der Rechner ist weiterhin für andere Zwecke einsetzbar, falls als ACC benutzt
- Zeiten können für Lohnabrechnung gewichtet werden
- individuelle Lohnabrechnung für jeden Mitarbeiter durch
- eigene Abrechnungsformel mit zeit- und lohnabhängigen Faktoren
- komplexe Berechnungen in Abrechnungsformularen

ADRESS DIREKT

Datensicherheit und Komfort

- geschützt durch Passwort
- Schutz der Installation vor Manipulationen wie Löschen etc.
- ‚private' Datensätze speicherbar, d.h. diese sind nur für den Anwender zugänglich, der sie gespeichert hat - der Superuser kann alle sehen beliebig viele Installationen speicherbar
- Zugriffsrechte für ‚Speichern', ‚Ändern', ‚Löschen',
- ‚private Datensätze', ‚Drucken', ‚Listenbearbeitung', ‚Installationserlaubnis' einzeln und für jeden Benutzer getrennt einstellbar
- jeder Installation kann ein eigener Pfad für den Dateizugriff vergeben werden
- Adressdaten nicht in einer Datei, sondern in mehreren verteilt, dadurch bei Diebstahl nicht reproduzierbar
- viele Einzeldaten kodiert abgespeichert
- jeder Installation, das heißt jedem Benutzer, kann eine eigene Textverarbeitung zugeordnet werden
- Start beliebiger Fremdprogramme möglich
- bis zu 7 Fenster mit beliebigen Inhalten
- großbildschirmfähig
- Zugriff auf die installierten ACC's
- bequemes Fensterhandling auch über Tastatur
- weit über den GEM-Standard hinausgehende Möglichkeiten der Dialogbearbeitung
- Popup-Menüs mit den wichtigsten Funktionen
- sofortiges Erkennen von Falscheingaben
- sofortige Feldformatierungen mit intelligenten Ergänzungsroutinen für Datum/Zeit editierbare Dateilisten in Listen geänderte bzw. gelöschte DS hervorgehoben
- Dateilisten abspeicherbar und wieder ladbar (stets mit aktuellen Werten!)
- schnelle Textsuche auch in Dateilisten unter Verwendung von Wildcards
- mächtige Listenfunktionen mit Rechnen im Text,
- Platzhalter für Datum, Uhrzeit, vielseitige Formatiermöglichkeiten
- verschachtelte Listenausgabe möglich
- schnelle Listenausgabe auf Laserdrucker, bei extrabreiten Listen auch gedreht
- flexibler Import (z.B. von BS-Handel-Daten, 1st-Address, Adimens, DBase)
- Netzwerkfähige Version für das PAM's Net lieferbar

XCOPY.DFÜ....ein Muß für alle Anwender mit mehr als einem ST – DFÜ-fähiges Kopierprogramm für RS232-, MIDI- und parallele Schnittstelle mit integr. Terminal und Kontrollfeld. Einsatzgebiet: Schneller Datentransfer von Rechner zu Rechner ohne Diskette; Anpassen der Inhalte versch. Festplatten an versch. Arbeitsplätzen bei automat. Sicherung bereits vorh. Daten; umfangreiche Selektionsmöglichkeiten der zu kopier. Dateien: Wildcards, Archivbit, autom. Backup etc; Versenden von Mitteil. von ST zu ST (Progr. arbeitet im Hintergrund) Einsetzbar als ACC, PRG (auch mit Parametern von Shells oder and. Programmen. Bitte INFO anfordern!

COMPUTER-PROJEKTIONSPANEL

Für alle, die etwas präsentieren müssen!

- Gestochen scharf, 8 Graustufen, hochmodernes DST
- LCD mit IR Fernbedienung!
- Keine Spezialsoftware nötig.
- Geeignet für jeden Overhead Projektor.
- Invertier-Funktion, Löschfunktion, auch an IBM PS/2, IBM PC XT/AT mit CGA, EGA, VGA-Karte an schließbar!
- Auflös. 640 x 480 Bildpunkte, nur 3 kg Gewicht!

Passender Metallkoffer 348,- **Preis 3798,-**

SOFTWARE

CAD
Cadja 998,-
Technobox Drafter 798,-
Technobox CAD/2 1998,-
Vorführung von kompletten CAD-Systemen unter MS-DOS oder auf Atari ST in unserem Beratungszentrum. Rufen Sie an!

DTP
Calamus 798,-
Calamus Font Editor 198,-
Outline Art 378,-
PKS-Write 198,-
Didot Fonteditor f. Calamus 198,-
Wir bieten Ihnen als DTP-Center individuelle Beratung für den Systemkauf im Bereich Desktop-Publishing. Rufen Sie an!

Textverarbeitung
1st Word Plus V3.15 inkl. 1st Address u. 1st Extra 239,-
1st Proportional Plus 119,-
Signum II 438,-
Scarabus Signum Fonteditor 98,-
Daily Mail 179,-
Writer ST 148,-
Word Perfect ST 748,-
That's Write Junior 148,-
That's Write 328,-
That's Write PostScript 498,-
Tempus 2.0 129,-
Tempus Word a.A.

Graphik
Creator 229,-
Imagic 458,-
Megapaint II 449,-
Megapaint II prof. 798,-
Omikron DRAW! 3.0 119,-
STAD 169,-
Retouche 399,-
Arabesque 268,-
Lavadraw 149,-

Datenbanken
1st Address 148,-
Adimen's 389,-
dbMAN V Version 5.2 incl. Comp., ISI.+Tempus 978,-

DFÜ
BTX-Manager incl. Interface 398,-
ohne Interface 278,-
Interlink ST 79,-

Tabellenkalkulation
BasiCalc 98,-
BasiChart 198,-

JAMES Das Börsenprogramm 298,-
GFA-Statistik 948,-

Kaufm. Software
Software BS-FiBu 548,-
Software BS-Handel 498,-
Software BSS Plus System a.A.
Software BSS Plus System (netzwerkfähig GTI-Netz) a.A.

Programmiersprachen
GFA-ASSEMBLER ST 139,-
Omikron Assembler 94,-
Omikron Macro-Assembler
GFA-Basic 2.0 EWS ST 44,90
GFA-Basic 3.0 EWS ST 189,-
GFA-Basic 3.5 EWS ST 40 neue Befehle (Vektor/Matrix) 268,-
GFA-Basic 68881 EWS ST 268,-
Omikron Basic 14,90
Omikron Basic 68881 Comp. 219,-
Omikron Basic Compiler 169,-
Omikron Basic Libraries
diverse Libraries ab 89,-
Omikron Basic Lib. SQL 288,-
Laser C 388,-
Prospero C 248,-
Turbo C 229,-
Turbo C + Debugger 448,-
Prospero Fortran 378,-
Prospero Fortran 68881 Lib. 198,-
SPC Modula-2 348,-
Prospero Pascal 248,-
GFA-BASIC KONVERTER nach C 478,-
GFA-CUP GEM UTILITY-PACKAGE 139,-

MIDI
Steinberg Cubase 790,-
Steinberg Twelve 99,-
EMC-Soundeditoren für verschied. Synthesizer/Expander
Roland D10/20/110 120,-
Ensoniq VFX 150,-
andere a.A.

Utilities
Harlekin
*ein ACC für alle Fälle 129,-
Turbo ST* Der Softwareblitter macht dem ST Beine 79,-
Querdruck * universell anwendbar für Tabellenkalkulation 79,-
BOOT IT * DAS Startprogramm für Festplatten 69,-

HARDWARE

Atari Mega ST1 mit 4 MB + SM124 + Maus 2998,-

Weide SCSI-Festplatten
Weide HD 50, 28 ns 1498,-
Weide HD 65, 28 ns 1698,-
Weide HD 85, 28 ns 1898,-
Weide HD 111, 19 ns Cache 2298,-
Weide HD 132, 19 ns Cache 2548,-
Weide HD 172, 19 ns Cache 3198,-
größere Platten a.A.
Alle Festplatten mit umfangreichen HD-Utilities, 100% komp. zu Atari-Festplatten, stabiles Metallgehäuse, ohne Lüfter!

Streamer
ICD-Streamer 155 MB 2498,-
Komfortable, schnelle u. zuverlässige Sicherung Ihrer Daten auf eine Datenkassette. Datensicherung mit 6.5 MB/Minute, durchgeführter und gepufferter DMA- und SCSI-Bus

Sub-Systeme
(*HD + Streamer in einem Gehäuse)
Weide HD 50 + Streamer 3798,-
Weide HD 111 + Streamer 4598,-
Weide HD 132 + Streamer 4998,-
Weide HD 172 + Streamer 5998,-

Floppy
3.5", anschlußfertig 289,-
5.25", 40/80 Tracks 448,-

Plotter
DIN-A3, Flachb., 8 Stifte 2451,-
DIN-A4 bis DIN-A0 a.A.
u.a. Grapftec A0-Plotter vorführbereit in unserem Beratungszentrum. Rufen Sie an!

Scanner
Printtechnik 200 dpi 948,-
Handyscanner 400 dpi, 105 mm 548,-
Bildverarbeitung auf Atari ST
Epson GT-4000 Farbscanner 8 Bit, professionell 4998,-

Emulatoren
PC-Speed 498,-
Supercharger 598,-
Spectre 798,-
Spectre GCR 1298,-

Modem
Discovery 2400 C 448,-
Discovery 2400 P Pocketmodem 449,-
Der Anschluß und Betrieb der Modems am Netz der DBP ist strafbar!

Weide Speicherkarten, die bewährten!
auf 1.0 MB für Atari 260, 520 348,-
auf 2.5 MB für Atari 260, 520, 1040 898,-
auf 2.5 MB für Atari Mega ST1 948,-
auf 4.0 MB für Atari Mega ST1 1548,-
auf 4.0 MB für Atari 260, 520, 1040 1498,-
Jede Erweiterung einzeln im Rechner getestet. Bebil-derte Einbauanleitung, einfacher Einbau ohne Löten. Durch vergoldete Mikrosteckkontakte optimale Schonung des MMU-Sockels. Kein Bildschirmflimmern, keine zus.Software, keine zus. Stromversorgung

Echtzeituhr Plus 129,-
Ab TOS '87 keine Boot-Software nötig. Jede Uhr im Rechner getestet und gestellt. Interner Einbau ohne Löten. Dadurch freier ROM-Port. Schaltjahrerkennung, Datum, Uhrzeit. Tausendfach bewährt!

Weide Coprozessor 68881
f. Mega ST incl. Software 498,-
Weide Coprozessor 68881 + WCL 898,-
Weide Coprozessor Language bringt Höchstleistung mit dem 68881 Coproz. INFO anfordern!

ICD Host Adapter
für Festplatten im Selbstbau
Anschluß von SCSI-Festplatten an Atari ST. Voller SCSI-Befehlssatz verfügbar. Umfangreiche Harddisk-Utilities.
Micro ST, die kompakteste Lösung für den Anschluß einer SCSI-Festplatte 249,-
Advantage ST, mit gepuffertem, durchgeschleiftem DMA-Bus 269,-
Advantage Plus ST mit zusätzl. Echtzeituhr 298,-

Digitizer
SAM-Echtzeit-Digitizer 998,-
Turbo-Dizer 348,-

Prommer
GAL-Prommer v. Maxxon 185,-
Junior Prommer 229,-
Rocke Eprommer 239,-

SIMM Speichererw. f. Atari STE
auf 2.0 MB 598,-
auf 4.0 MB 1198,-
Blitter Nachrüstplatine 89,-
Eprombank 64 KB 19,80
Epromdisk 512 KB 139,-
Atari Trackball f. Atari ST 98,-
kabellose Infrarot-Maus 198,-
Marconi Trackball, d. Professionelle 198,-



Preiswertes Farbband

Farbbänder für Drucker sind meistens zu schnell aufgebraucht und dann auch noch viel zu teuer. Wer sich nicht davor scheut, auch mal schwarze Finger zu bekommen (die Zeit zum Waschen fehlt ja fast immer), kann auf einfache und preiswerte Art zu einem verhältnismäßig guten "Fast-Karbon-Ausdruck" kommen. Die Preise für Farbbänder sind ja hinreichend bekannt. Aber die Herstellerangaben über die Lebensdauer wagt man zu bezweifeln. Da wird z.B. von 200000 Zeichen gesprochen, was in etwa 40 DIN A4-Seiten entspricht (man möge mich korrigieren), und das für 30,- bis 40,- DM. Nun mein Tip, der möglicherweise gar nicht ankommt, weil alte Hasen schon selbst darauf gekommen sind:

Kaufen Sie sich einen Haufen Kohlepapier im Schreibwarengeschäft. 500 Blatt kosten maximal DM 200,- und sind sogar noch mehrfach verwendbar. Doch auch wenn Sie ein Blatt nur einmal benutzen, zahlen Sie für die oben genannten vierzig Ausdrucke nur zirka DM 16,-. Getestet habe ich es mit einem Epson LQ-550 mit Papierstärkeneinstellung 5-7. Natürlich muß man vorher das Farbband herausnehmen, was ja kein Problem ist. Das Kohlepapier kommt (sinnigerweise) mit der Schichtseite auf das zu bedruckende Papier. Wie Sie den Packen nun in den Drucker einlegen, ist Ihre Sache. Sie werden schon sehen, ob der fertige Ausdruck seitenrichtig herauskommt. Ein wichtiger Hinweis noch: "Nich mitte Finger auffe Dinger packen!", denn sonst sieht's wirklich nach Kohle aus. Und zum Schluß sollte man nicht vergessen, das herausgenommene Farbband wieder einzulegen, denn ohne das geht die normale Arbeit auch mit dem bestgemeinten Tip nicht weiter.

Klaus Tschentscher

STe-Kompatibilität überlisten

Beim "Ausprobieren" eines 1040 STe mußte ich feststellen, daß einige Programme auf diesem Gerät zunächst nicht laufen. Nach einigem Experimentieren habe ich jedoch folgende Lösung für das Problem gefunden: Auf dem STe ist auch das alte Disk-TOS vom 6.2.1986 (Version 1.0) lauffähig. Bootet man den STe mit einer solchen TOS-Systemdiskette, sind einige Programme wieder lauffähig, die sich zwar nicht mit dem TOS 1.6 vertragen, sehr wohl aber mit dem TOS 1.0. Das trifft z.B. auf verschiedene Spiele und Anwendungsprogramme zu. Inwieweit dies auch mit einer TOS 1.4-Version klappt, kann ich in Ermangelung einer solchen nicht sagen.

Anmerkung der Redaktion: Alle TOS-Versionen können auch auf dem STe per Diskette eingeladen werden. Dadurch lassen sich verschiedene Kompatibilitätsprobleme beheben.

Klaus Opel

Tastaturtip

Hier ein Tastaturtip zu "Funckey" (AUGE 11): Bei Belegung der Funktionstasten durch Sonderzeichen kann es durch die Unterschiede der deutschen und der amerikanischen Tastatur zu Problemen kommen. So muß z.B. zum Ausgaben des "/"-Zeichens die Taste "&" gedrückt werden. Allgemeines Vorgehen: Suche auf der Tastatur das Zeichen, das Du ausgeben willst. Suche diese Taste auf der amerikanischen Tastaturschablone (Benutzerhandbuch) und gib das entsprechende Zeichen auf der deutschen Tastatur ein. Beispiele:

```
beim / das & drücken
beim : das > drücken
beim y das z drücken
beim ( das * drücken
```

Der Tastaturtip für die amerikanische Tastaturbelegung: Wer bei Programmen mit amerikanischer Tastaturbelegung auf die Schnelle ein "ö", "ä" oder ein "ü" braucht, muß zuerst Alt-k drücken (noch sieht man nichts) und dann die entsprechende Taste (o, a oder u). Nun erscheint z.B. ein ö statt eines o. Um ein "ß" zu erhalten, muß man die Tastenkombination Alt-s drücken.

Christoph Ludwig

Formulare mit 1st_Wordplus

Wer mit 1st_Wordplus Formulare ausfüllen will oder muß, hat damit seine liebe Not, denn 1st_Word unterstützt keine Maskentechnik. Wer einen Font-Editor, vielleicht sogar Harlekin besitzt, kann sich jetzt mit einfachen Mitteln behelfen. Das Problem ist, daß 1st_Word alle Zeichen, die auf dem Bildschirm sichtbar sind, auch ausdruckt (jedenfalls das alte 1st_Word. Das neue druckt auch Zeichen, die NICHT auf dem Bildschirm sichtbar sind!). Somit dürfen Felddeklarationen wie z.B. Name, Vorname usw. nicht im Dokument stehen oder müssen vor jedem Ausdruck mühselig gelöscht werden.

Ich habe mir jetzt so geholfen, daß ich mit Hilfe eines Font-Editors in dem ASCII-Bereich $B4-$DC (hier liegen sonst die hebräischen Zeichen) noch einmal einen Satz Ziffern, ein großes ABC und einige Sonderzeichen (.:_%/) definiert habe. Diese Zeichen unterscheiden sich optisch von den normalen Zeichen. In meinem Fall habe ich die Linien nur 1 Pixel stark definiert. Um diese Zeichen von 1st_ Word aus zugänglich zu machen, muß man den Druckertreiber in dem o.g. Bereich anpassen, dort wird für alle neudefinierten zeichen ein Space-Zeichen ($20) eingesetzt, z.B. B4, 20 oder B5, 20 usw.

Nach der Installation des neuen Treibers und Fonts können jetzt die neuen nichtdruckbaren Zeichen durch Anklikken in der Auswahlbox vom 1st_Word in den Text übernommen werden. Das ist nicht komfortabel, aber ein Formular wird nur einmal erstellt und kann als Format abgespeichert werden. Harlekin-Besitzer haben es noch besser: Sie verändern zur Maskenerstellung ihre Tastaturbelegung und legen die nicht zu druckenden Zeichen einfach auf das Tastatur-Alphabet. So können sie ihre Maske flüssig schreiben.

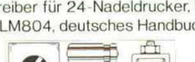
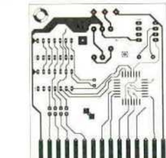
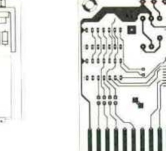
R. Gottberg

Take 20

Hard- und Software

Take 20 - Tip des Monats: 20

PCB-Edit V2.0 **DM 199,-** *
Platinenlayoutprogramm für den Atari ST. Hohe Auflösung (180 DPI), WYSIWIG, 12 Lötaugen, 6 Leiterbahnen, SMD, Feinstleitertechnik, Masseflächen, Beschriftung, Bibliotheken, Bohrplan, Bestückungsdruck, Schaltpläne usw. Treiber für 24-Nadeldrucker, HP-Laserjet und Atari SLM804, deutsches Handbuch, kein Kopierschutz

Skizzenbücher von Volkmar Hoppe entdeckt! 11

Wer mit Text- und DTP-Programmen arbeitet, braucht hochwertige und aussagekräftige Grafiken zur copyright-freien Verwendung: Der Künstler Volkmar Hoppe hat seiner Phantasie freien Lauf gelassen und unzählige reale wie irreale Assoziationen in seine Arbeit einfließen lassen. Bildhafte Wortspiele, satirische Anmerkungen, zynische Reflexionen, Sinnvolles, aber auch Spielerisches und Witziges stellen das künstlerische Ergebnis dar, das mit geübter Feder zu Papier bzw. auf die Diskette gelangt ist. Das komplette Paket umfaßt 17 Disketten.



Brainstorm 4 by Hoppe **DM 189,-** *
120 Seiten DIN A4 auf 17 Disketten. Alle Dateien im GEM Image-Format: In Calamus, 1st Word und ähnlichen Programmen direkt verwendbar. Handbuch und Konvertierungshilfe.

Privatliquidation / Praxibuchhaltung 1
Aus der Praxis für die Praxis, **je DM 375,-** *
ReProk **DM 598,-** *
Date Access **DM 49,-** *
Universeller Terminplaner mit Adressenverwaltung, Notizblock, Suchfunktionen uvm...
NotDatMan **DM 198,-** *
Noten/Daten-Manager für Lehrer. Komfortable Notenverwaltung, Auswertung von Korrekturlisten, Absenzenlisten, Verwaltung von Schülerdateien und zahlreiche Statistikfunktionen.

BILDBANK **DM 139,-** * 2
Komfortables Informationssystem zur Verwaltung von Bilddaten im Degas, NEO, ART und IMG-Format. Bilder, Begleittexte und Bedienungshilfe in separaten Fenstern. Beliebig viele Bilder in der Datenbank (z.B. 1000 NEO-Bilder auf 30MB Festplatte). Läuft auf allen Monitoren, auch auf Großmonitor und Big-Screen.
ADRESSWRITER **DM 149,-** *
Adressenverwaltung mit eigenem Texteditor für Serienbriefverarbeitung. Adressdaten können über ACC-Funktion in andere Programme eingefügt werden. Flexible ASCII Import-/Exportfunktionen, Telefontaktzähler für In-/Auslandsgespräche und noch vieles mehr...

BASiCHART 1.0 **DM 198,-** * 4
Schnelle und komfortable Tabellenkalkulation mit integrierter Präsentations-Graphik.
BASiCALC 2.0 **DM 98,-** *
LDW Powercalc **DM 249,-** *
Logistix **DM 249,-** *
VIP Professionell **DM 149,-** *
Querdruck 2 **DM 78,-** *
Druckt Texte bis 32000 Zeichen Breite. Jetzt mit neuer Oberfläche, Tutorial, neuen Schriften, Zeichensatzeditor, verbessertem Einzelblatteinzug, als Accessory, WP-Modus aus WordPlus und Ausgabe auf ATARI-Laser.

STEVE 3 **DM 498,-** * 6
Integriertes Programm mit Textverarbeitung, Datenbank, Grafik, Deskop Publishing und CAI (Computer Aided Instructions). Einfache Serienbrieferstellung, Textbausteine, Rechenmöglichkeiten im Text, Übersetzungsfunktionen, Datenbank mit der Möglichkeit Bilder einzubinden, mischen von Text und Grafik, Formblatterstellung, Wecker, und vieles mehr.
STEVE-Extra **DM 68,-** *
Lehrbuch zu STEVE, inklusive Beispieldiskette.

READPIC **DM 150,-** * 6
Lernfähiges Texterkennungssystem für Cameron Handyscanner und Geniscan. Bedienung vollständig unter GEM, hohe Erkennungsrate, Speichern des erkannten Textes im ASCII-Format und anschließender Import in Textverarbeitungs- und DTP-Programme.
Signum!2 **DM 418,-** *
Script **DM 198,-** *
Scarabus **DM 95,-** *
Fonteditor für Signum! und Script.
That's Write Junior **DM 148,-** *
That's Write **DM 328,-** *

Lern ST plus **DM 59,-** * 8
Maskenorientiertes Universal-Lernprogramm für Sprachen, Geschichte und Faktenwissen aller Art. Komfortabel und leistungsfähig.
Zusatzdisketten für Lern ST plus, je DM 20,- *

Englisch	(3000 Vokabeln)
Englisch 2	(3000 weitere Vokabeln u. Wendungen)
Französisch	(4500 Vokabeln und Wendungen)
Spanisch	(4500 Vokabeln und Wendungen)
Italienisch	(ca. 4000 Vokabeln und Wendungen)
Latein	(ca. 3000 Vokabeln und Wendungen)
Geschichte	(ca. 300 wichtige Ereignisse)
Führerschein	(ca. 400 Multiple-Choice-Fragen)

STransPlus **DM 99,-** * 8
Elektronisches Wörterbuch mit 20.000 englischen Vokabeln. Per ACC-Funktion auch aus Textprogrammen bequem erreichbar. Die Integrierte Übersetzungshilfe liest ASCII-Text ein und zeigt gefundene Übersetzungen eines Wortes an. Einfache "Roh"-Übersetzungen (ohne grammatikalische, syntaktische oder inhaltliche Überprüfung) erstellt STransPlus auf Wunsch automatisch. Mit Vokabeltraining.

DR. SCHELM **DM 59,-** * 8
Fesselndes Quizprogramm für die ganze Familie. Update-Version: jetzt mit 2-Spieler-modus (Doppelschelm), 600 Fragen aus 30 Wissensgebieten und zusätzlicher Spielvariante "Super-Hangman". Lehrreiche und humorvolle Unterhaltung.
Zusatzdiskette zu DR. SCHELM **DM 15,-** *
Enthält weitere 600 Fragen aus 30 Gebieten.
Erdkunde 3.0 **DM 69,-** *
ST-Math **DM 98,-** *

Neue Spiele

Adventures (4 Spiele: Emmanuelle, Freedom, 20.000 Meilen unter dem Meer und African Raiders)	84,- *
Austerlitz 1805	79,- *
Conquerer 3D	84,- *
Cyberball	59,- *
Dragons Breath	94,- *
Dragons Lair	139,- *
Frontline	79,- *
Full Metal Planet	84,- *
Gold of the Americas	79,- *
Operation Thunderbolt	64,- *
Sherman M4	84,- *
SIM-City	94,- *
Tower of Babel	79,- *
Ultima 5	94,- *
West Phaser 1870 - Ballerspiel mit Lichtpistole	119,- *
Windwalker	94,- *
Freedom Connection Infrarot-Fernbedienung für Joysticks. Spielen ohne lästigen Kabelsalat	79,- *

Die vorgestellten Produkte erhalten Sie bei einem der folgenden Anbieter:

HD Computertechnik
Pankstr. 42
1000 Berlin 65
030 / 4657028-29

T.U.M. Soft+Hard
Hauptstr. 67
2905 Edewecht
04405 / 6809

Computer Technik Kieckbusch GmbH
5419 Vielbach
02626 / 78336 (Fax: 78337)

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90
069 / 763409

IDL Software
Lagerstraße 11
6100 Darmstadt 13
06151 / 58912

Wir suchen Programmierer ...
zur Weiterentwicklung von Treibersoftware für Video-Digitizer. Hardware und Source-Material wird gestellt.
Wir suchen Grafikspezialisten und Playtester ...
für verschiedene Programmprojekte und Testberichte im Strategie- und Rollenspielbereich.
Bewerbungen und Angebote senden Sie bitte an: PD-Szene, Postfach 130104, D-6100 Darmstadt.

CADjA V.1.2 **DM 998,-** * 10
CAD-Programm der Spitzenklasse, anwendbar für Elektronik, Maschinenbau, Architektur. Plotausgabe bis DIN A0. Symbolbibliotheken verfügbar. ASCII- Schnittstelle. Direkte Weiterverarbeitung mit "CADjA-CAM". Schnittstelle zu STEVE. Leistungsmerkmale: überragend! Druckausgabe auf Nadeldrucker, Laser, Plotter. Demo: DM 50,- *
STar Designer **DM 149,-** *
ARABESQUE **DM 278,-** *

CALAMUS 1.09 **DM 799,-** * 11
OUTLINE ART **DM 398,-** *
Reprostudio ST **DM 498,-** *
Retouche **DM 399,-** *
PKS-Write **DM 198,-** *
CALAMUS Fonteditor **DM 198,-** *
DIDOT Fonteditor **DM 198,-** *
Layout-Paket für Calamus **DM 79,-** *
Gestaltungshilfe, Pass- und Schnittmarken in DIN Standartformaten A5, A4 und A3, hoch u. quer. Mit Anleitung (auf Wunsch in Englisch).

Architektur & Design **DM 59,-** * 11
Über 400 Vektor- und Rastergrafiken (GEM-/PAC-Format) für CALAMUS.
Geben Sie Ihrem Calamus die Fonts, die er braucht!
Plup, Bobble, Skript, ROCA, YAPPIE, Caslo, ROMA REG, AltBerlin bold, CARDPLAY, Peking reg, Rund JILLY, INFRA, AKTIVA, GEODET, Boedet

Layout-Paket für Calamus **DM 79,-** * 11
Gestaltungshilfe, Pass- und Schnittmarken in DIN Standartformaten A5, A4 und A3, hoch u. quer. Mit Anleitung (auf Wunsch in Englisch).
Marconi Trackerball **DM 198,-** *
100%tiger Mausersatz mit hoher Lebensdauer.
VIDI-ST **DM 398,-** *
Echtzeit-Framegrabber für alle ST. Digitalisiert Videobilder in 16 Graustufen (20ms/Bild, Low-Res.); speichert Bildfolgen (12 Bilder/sec) ins RAM. Animation mit bis zu 25 Bildern/sec. Verarbeitet NEO-, Degas- und IMG-Format.

VOMBLATT **DM 89,-** * 12
Musikdidaktisches Programm zum Erlernen des Notenlesens - auch ohne Vorkenntnisse - bis Hochschulniveau. Alle Schlüssel, Ein-/Ausgabe über MIDI möglich, Prüffunktionen und mehr...
Steinberg 'Twelve' **DM 99,-** *
12 Spur Midi Sequencer. 'Twelve' ist der kleine Bruder des schon populär gewordenen Steinberg Twenty-Four.
Cubase **DM 790,-** *
Desktop Midi Recording System

Love&Fun zeigt, welcher Partner zu Ihnen paßt: In der Liebe, beim Sport und am Arbeitsplatz. Kern der Profi-Version ist die schnelle Adressverwaltung, deren gesamter Datenbestand zur Partnersuche herangezogen werden kann. Mächtige Analysemöglichkeiten, flexible Suchkriterien, einfache Bedienung und die elegante Oberfläche zeichnen Love&Fun aus. **DM 59,-** * 13
Dr. Lustig **DM 29,-** *
Der Psychoanalytiker. Ein toller Partyspaß.

ST-DCL **DM 298,-** * 14
Kommandozeileninterpreter + Tools ST-DCL erlaubt es, auf einem Atari ST unter einer Kommandoumgebung zu arbeiten, die der VAX/VMS-Digital Command Language (DCL) nachempfunden ist. Auch für MS-DOS lieferbar.
ST Pascal plus **DM 249,-** *
SPC Modula 2 **DM 348,-** *
STAN **DM 69,-** *
Zur statistischen Auswertung und grafischen Darstellung von Zahlen und Messreihen.

Desk Assist 4.2 **DM 188,-** * 19
Das Multifunktions-Accessory! Terminplaner mit Alarm- und Dauerterminen, Adressen- und Telefondatei für Serienbriefe, Druckerspooler, Taschenrechner, Maßumrechnung, RAM und Diskeditor, ASCII-Tabelle und noch mehr ...
SIM Rechner-Freezer **DM 35,-** *
Speichert den aktuellen Betriebszustand des ST auf Diskette ab. "Eingefrorene" Programme können schnell und einfach reaktiviert werden.
Quick-Screen **DM 49,-** *
Beschleunigt Bildschirmaufbau u. Textausgabe.

5.25" Laufw., 40/80 Trks. **DM 318,-** * 20
3.5" NEC Laufwerk **DM 259,-** *
Slim-Line, anschlußfertig und durchgeschliffen.
EL 3.5" Laufwerk **DM 248,-** *
EX 30L Festplatte **DM 1298,-** *
EX 60L Festplatte **DM 1998,-** *
EX 40W Wechselplatte **DM 2698,-** *
MEGA-DRIVE 40A **DM 1398,-** *
Einbauplatte 40MB für alle MEGA ST.
ST-Uhr 2 für 520/1040ST **DM 79,-** *
Auf Soundchip steckbar, einfacher Einbau.
EM 124 Multi-Monitor (f & s/w) **DM 598,-** *

RAM-Erweiterung 2,5 MB **DM 898,-** * 20
Mit 4 MB Speicherkarte (steckbar) von Weide.
RAM-Erweiterung 4 MB **DM 1498,-** *
Wie oben, jedoch mit vollen 4 MB bestückt.
A1MB Ram-Erweiterung **DM 548,-** *
A3MB Ram-Erweiterung **DM 1148,-** *
Neu: Der bereits vorhandene Speicher Ihres ST bleibt einsatzbereit. Sie benötigen somit nur A3MB zur Aufrüstung von 1 MB auf 4 MB.
MiniRAM 1MB-Erweiterung **DM 298,-** *
4-Bit organisierte Erweiterung für 260/520 ST.

Rosin Datentechnik
Peter-Spahn-Str. 4
6227 Oestrich-Winkel
06723 / 4978

KREATIV-Software
Oberwürzbacher Str. 10
6676 Mandelbachtal
06803/3850 u. 06805/2666

Musikinstrumente&Computer
August-Bebel-Str. 3
6840 Lampertheim 5
06241 / 80899

Weeske Computer
Potsdamer Ring 10
7150 Backnang
07191 / 1528-29 od. 60076

Duffner's PD-Center
Ritterstr. 6
7833 Endingen a.K.
07642 / 3875 od. 3739

T.S. Service
Szemere Hard & Software
Schleißheimerstr. 127, 8 Mü 40
089 / 3089408

Dietmar Schramm
Promberg 6
8122 Penzberg
08856 / 7287

Schick EDV-Systeme
HauptstraßeO 32a
8542 Roth
09171 / 5058-59

PR8-SOFT, Klaus-M. Pracht
Otto-Hahn-Str. 10
8702 Estenfeld
09305 / 8211

Bossart-Soft
Sonnenhofstr. 25 / PF 5146
CH-6020 Emmenbrücke 3
041 / 458284

* Alle Preise sind unverbindlich empfohlene Verkaufspreise

Coupon

Ich bestelle:

Name, Vorname

Straße

Plz, Ort STC 7-8/90

☐ Per Scheck. Zuzüglich DM 5,- Versandkosten, unabhängig von der bestellten Stückzahl.
☐ Per NN (Nur Inland, zuzügl. DM 8,- NN-Gebühr)

Unser ProFile-System – Ihr Vorteil.



Unser ProFile-System bietet ihnen die Zuverlässigkeit und die Flexibilität, die Sie als Atari-User heute brauchen. Spitzenentwickler waren an der Arbeit – das Ergebnis sind Geräte mit den besten Eigenschaften.

Festplatten von 20 bis 180 MB Speicherkapazität für eine schnelle – und leise – Verarbeitung und Verwaltung von Daten. Die DC-Serie ist zusätzlich mit einem integrierten 64-KB-Cachespeicher ausgerüstet, der die Zugriffszeit um bis zu 50 % verringert.

protar **– eine sichere Entscheidung für die Zukunft**

Bis zu 2 Jahre Garantie gibt die notwendige Sicherheit. Zu unserer Produktpalette gehören ebenso wiederbeschreibbare optische Speicher und Monochrome-Bildschirme sowie das ProFile R44 – ein Wechselplatten-System – mit dem Sie 44 MB schnell zur Hand haben.

Informationen zu protar-Produkten erhalten Sie bei Ihrem qualifizierten Fachhändler.

protar Elektronik GmbH · Puttkamer Straße 7 · D - 1000 Berlin 61 · Telefon 030 - 251 30 65 · Telefax 030 - 251 30 68
Vertretung in Österreich: Dipl.-Ing R. Temmel Ges. m.b.H. & Co. KG · Markt 109 · A - 5440 Golling · Telefon 06244 - 70 81 0 · Fax 06244 - 71 88 3
Vertretung in der Schweiz: DTZ DataTrade AG · Landstraße 1 · CH - 5415 Rieden/Baden · Telefon 056 - 82 18 80 · Telefax 056 - 82 18 84



Ein Wort in eigener Sache

In den Jahren, die unsere Zeitschrift existiert, haben wir immer wieder versucht, durch die Beantwortung der bei uns eingehenden Briefe ein wenig Licht in das Dunkel zu bringen, das bei der Arbeit mit dem ATARI ST schon so manch einen aus der Fassung bringen konnte - eine Tatsache, die nicht nur Ihnen, verehrter Leser, sondern auch uns oft genug zu schaffen machte. Nichtsdestotrotz haben wir uns bemüht, die Probleme zu lösen und diverse Leserbriefe zu veröffentlichen, da wir der Meinung waren, daß die jeweilige Thematik auch einen größeren Leserkreis interessieren könnte. Trotzdem gibt es immer wieder Briefe, die wir nicht beantworten können oder dürfen. Damit Sie nicht allzusehr enttäuscht zu sein brauchen oder keine Antwort erhalten, möchten wir Sie bitten, sich an folgende Spielregeln zu halten, die sich aus unserer Erfahrung ergeben haben. Fällt Ihr Brief nicht unter die folgenden Kriterien, hat er gute Chancen, positiv beantwortet oder wenigstens als Hilferuf an unsere Leserschaft gedruckt zu werden.

1. Leider gehen immer wieder Briefe mit dem Wunsch ein, ein Produkt für diesen oder jenen Anwendungsfall vorzuschlagen, verschiedene Produkte bezüglich der Vor- und Nachteile gegeneinander abzuwägen und zu bewerten. Es ist uns aus Wettbewerbsgründen nicht erlaubt, ein bestimmtes Produkt zu favorisieren, selbst wenn wir das eine oder andere in der Redaktion überzeugt einsetzen. Wir können Sie in diesem Fall ausschließlich auf die von uns möglichst objektiven Tests und eventuell anstehende Fachmessen hinweisen. Bedenken Sie bitte, daß auch wir nicht jede Textverarbeitung, jedes Malprogramm und so weiter kennen und bestimmte Produkte dadurch in das Abseits drängen würden.

2. Oft erreichen uns Briefe, die sich positiv oder auch negativ über bestimmte Händler, Softwarehäuser oder deren Produkte auslassen. Sicherlich interessieren uns solche Bemerkungen. Bitte haben Sie aber Verständnis, daß wir weder Lob noch Tadel abdrucken dürfen, da diese Aussagen meist subjektiv sind. Anders sieht die Sache beispielsweise bei Gerichtsurteilen aus, die Sie, verehrte(r) Leser(in), erfochten haben.

3. Aufgrund der Vielzahl an Briefen, die uns täglich erreichen, sind wir leider nicht in der Lage, Programmfehler anhand von Listings oder ähnlichem zu korrigieren. Dennoch sollte ein Problem möglichst detailliert beschrieben sein, denn Ferndiagnosen sind prinzipiell sehr schwer, jedoch mit genauerer Angabe der Symptome eventuell durchführbar.

4. Von Zeit zu Zeit erreichen uns Briefe mit der Bitte, die Adresse des Lesers zwecks allgemeiner Kontaktaufnahme zu veröffentlichen. Würden wir dies in die Tat umsetzen, würde sich der Umfang des anderen redaktionellen Teils beträchtlich verkleinern. Ausnahmen stellen Leser in fernen Ländern dar, für die eine Kontaktaufnahme im eigenen Land recht schwierig ist.

Zum Schluß sollen ein paar Tips eventuell voreilig geschriebene Briefe verhindern.

1. Wenn Sie ein Problem bezüglich einer bestimmten Problematik haben oder an einem bestimmten Produkt interessiert sind, finden Sie interessante Artikel darüber eventuell in vorhergehenden Ausgaben userer Zeitschrift. Zur Auswahl eignet sich das Jahresinhaltsverzeichnis besonders gut, das immer am Jahresende in der ST Computer abgedruckt wird.

2. Sollten die Probleme mit der Handhabung eines Produktes zu tun haben, wenden Sie sich zunächst an Ihren Händler und über diesen an den Distributor beziehungsweise an das Software-Haus. Die Wahrscheinlichkeit, daß Ihnen das Software-Haus weiterhelfen kann, ist um ein Vielfaches höher als die, daß wir Ihnen helfen können.

3. Lesen Sie aufmerksam die Leserbrief-Seite. Viele Fragen wiederholen sich immer wieder, obwohl wir bestimmte Probleme schon mehrfach angesprochen haben.

Dirsleft?

Auf Empfehlung der Firma Atari wende ich mich an Sie mit der Frage, ob Sie mir mit den von Ihnen vertriebenen PD-Programmen bei der Lösung folgender Software-Probleme behilflich sein können:

a) Bei der Festplatte SH205 vermisse ich das auf der mitgelieferten Diskette "Atari Hard Disk Utilities" das in der Literatur erwähnte Programm DIRSLEFT.PRG zur Ermittlung der noch zur Verfügung stehenden Directories.

b) Beim Ausdruck einer ASCII-Datei auf einem HP DeskJet erscheint für das auf dem Bildschirm sichtbare "ß" ein "Pts". Wie kann ich das verhindern?

c) Eine Hardcopy läßt sich nicht ausdrucken. Es erscheinen nur wirre Zeichen. Gibt es hierfür einen speziellen Druckertreiber?

Gerd Goldstein, Köln

Red.: Es ist schön, daß die Firma Atari Sie an uns verwiesen hat (!), das macht sie sehr oft. Früher wurde das Programm "Dirsleft" auf der SH205-Diskette ausgeliefert. Wenn es auf Ihrer Diskette fehlt, sollten Sie sich an Ihren Händler wenden. Daß der HP DeskJet ein Pesetas-Zeichen ausdruckt, liegt daran, daß beim Atari ST das scharfe S den ASCII-Code 158 besitzt, während fast alle anderen Rechner es bei ASCII 225 positioniert haben. Durch einen entsprechenden Druckertreiber (z.B. mit 1st_ Wordplus) läßt sich das entsprechend verhindern. Die andere Möglichkeit ist, für ein scharfes S immer den Buchstaben mit dem Wert 225 einzugeben, was aber nur mit einer Umdefinierung der Tastatur möglich ist (die z.B. mit dem Programm Harlekin funktioniert). Für eine Hardcopy gilt das gleiche: Sie benötigen dazu einen entsprechenden Druckertreiber.

Pro und Contra Speichererweiterung

Lieber Kummerkastenonkel! Seit vier Monaten bin ich mehr oder weniger zufriedener Besitzer eines 1040 STe. Bei der Anschaffung war von vornherein klar, daß es hauptsächlich im Bereich CAD zum Einsatz käme. Dieses brachte ich gegenüber dem Verkäufer klar zum Ausdruck, der mir den heiligsten Pfadfindereid schwur, daß ein RAM-Speicher von 1 MB auch für umfangreiche technische Zeichnungen mehr als ausreichend sei. Alles Lüge! Schon nach kurzer Einarbeitungszeit stieß ich an die Grenzen der Speicherkapazität. Verständnisvolle, wohlwollende Händler, die mit derartigen problemen vertraut zu sein scheinen, bieten als Lösung Speichererweiterungen an. Hier nun die Masterfrage: Wie betriebssicher sind solche Speichererweiterungen? Einige Händler versicherten mir, eine Speichererweiterung sei völlig problemlos und außerdem das Größte seit Erfindung des Rads. Andere erklärten mir mit vor Entsetzen verzerrtem Gesicht, sie lehnten derartige Speichererweiterungen grundsätzlich ab. Angegebener Grund: ständige Systemabstürze! Von TOS-Fehler 35 war die Rede. Wem soll ich nun glauben, dem, der da sagt "Null Problemo!" oder dem, der meint "il catastrophico!"?

Stefan Schulze, Oldenburg

Red.: Schicken Sie den Händler zum Teufel, der Speichererweiterungen ablehnt. Er will Ihnen lediglich ein neues Gerät verkaufen und damit mehr Geld verdienen. Gerade beim STe ist eine Speichererweiterung kein Problem: Sie müssen nur passende SIMM-Karten einstecken und die Erweiterung ist vollbracht. Leider kann es aber sein, daß in Ihrem STe bereits 4 Karten stecken,

die insgesamt 1 MB ergeben. Wenn Sie hier auf 2 MB erweitern wollen, können Sie das vorhandene Megabyte vergessen und müssen 2 neue MB kaufen. Sollten Sie nur eine SIMM-Karte mit 1 MB haben, benötigen Sie nur eine weitere mit einem Megabyte. Der TOS-Fehler 35 tritt normalerweise nur dann auf, wenn versucht wird, eine korrupte Programmdatei zu starten.

*

STe inkompatibel?

Ich bin seit kurzem Besitzer eines Atari 1040 STe. Leider mußte ich erfahren, daß meine Maschine nicht 100% kompatibel zu anderen STs ist. Da ich das Risiko, Programme für gutes Geld zu kaufen, die nicht laufen, nicht eingehen möchte, habe ich folgende Frage: Welche guten Spiele laufen auf dem Atari 1040 STe einwandfrei?

Roland Heim, Bregenz

Red.: Leider können wir Ihnen keine Liste aller funktionierenden Programme für den STe präsentieren, da die Liste den Umfang einer Ausgabe sprengen würde. Am besten sollten Sie das entsprechende Programm bei Ihrem Fachhändler ausprobieren.

*

XF551-Floppy

Seit 4 Monaten arbeite ich mit einem 520 ST. Vorher hatte ich einen Atari 130 XE mit der Floppy XF551. Ich möchte meine Programme vom XE auch auf dem ST weiterverwenden. Ist es also möglich, die Floppy XF551 am ST zu verwenden und wenn ja, wie bzw. wer nimmt den Umbau vor? Gibt es einen Emulator für den ST, unter dem die XE-Programme laufen (möglichst als PD-Software)?

Daniel Deistler, DDR-Blankenhain/ Thüringen

Red.: Es ist nicht möglich, das alte Laufwerk am ST zu betreiben. Es existiert allerdings ein XL-Emulator, mit dem teilweise auch die XE-Programme zusammenarbeiten müßten. Dieser Emulator ist nach unseren Informationen sogar PD.

*

C64 zum ST

Im März dieses Jahres brachten Sie in der Rubrik Programmierpraxis einen Beitrag von Helmut Büche mit dem Titel "Commodore 1581-Wandler". Auch ich besitze neben meinem ST einen 64er, mit dem ich immer noch gelegentlich arbeite. Vor allem aus vergangenen Computer-Jahren habe ich noch ein paar Dateien auf dem 64er, die ich gerne mit meinem ST weiter nutzen würde. Leider besitze ich kein 1581-Laufwerk, sondern noch eine alte 1541-Floppy. An meinem ST habe ich allerdings eine 5,25"-Floppy angeschlossen - die Diskettengröße spielt also keine Rolle. Gibt es irgendeine Möglichkeit, Commodore-Disketten mit diesem Laufwerk einzulesen? Ich habe bereits versucht, mit einem Kopierprogramm 64er-Disketten auf 3,5"-Disks zu kopieren - bisher aber ohne Erfolg.

Edgar Dehm, Nürnberg

Red.: Die direkte Übertragung einer 1541-Diskette ist nicht möglich, da die 1541 ein komplett anderes Diskettenformat besitzt als die 1581 - man sieht es bereits an der Anzahl der freien Blöcke auf einer Diskette. Die einzige Möglichkeit, alte Dateien weiterzubenutzen, wird die Überspielung per Nullmodem-Kabel sein. Dieses können Sie beispielsweise bei Mükra oder Resco kaufen, oder es selbst bauen. Die genaue Bauanleitung erfahren Sie am besten bei Arnd Wängler, einem Redakteur der 64'er, unter der Telefonnummer (089) 4613-202.

Vektorverbieger

Zu Weihnachten bekam ich etwas für den Golfspieler, "der schon alles hat": vergoldete Tees aus Christbaumkugelglas. Unwillkürlich mußte ich einen Vergleich ziehen. Was kann ein Programmierer für die Zeitschrift ST Computer denn noch programmieren? Viele haben schon MULTIFIX Ver. 3.09, RAPIDO Ver. 2.18 und Emulatrix IV. Schon beim passenden Namen wird es schwer, die halbe Götterwelt ist bereits besetzt und anspruchsvoll soll der Name doch sein! Ach, was bin ich als Leser froh, daß ich kein Vektorverbieger bin!

Rudolf Burkhardt, Bremerhaven

Red.: Sie haben recht: Tatsächlich ist die halbe Götterwelt bereits mit Namen belegt. Allerdings sollten sich einige Programmierer besser überlegen, ob sie ihren Programmen nicht lieber einen passenden Namen geben sollten, der sofort aussagt, was man damit anfangen kann und was nicht. In der Tat sagt ein Programmname wie "Aphrodite" oder "Caesar" recht wenig aus - Umdenken ist angesagt. Allerdings sollte man auch nicht alle Programmnamen in der ST Computer allzu wörtlich nehmen...

*

Sozobon C

Ich benutze seit einiger Zeit den Public Domain C-Compiler Sozobon C. Dabei sind mir zwei Fehler aufgefallen. Das Compiler-Programm HCC (V1.01) hängt sich auf, wenn ein Kommentar nicht abgeschlossen wird. Beim Durchsehen der Quelltexte zu HCC war es nicht schwer, den Fehler zu lokalisieren. In der Datei TOK.C gibt es eine Routine namens *dospec()*, an deren Anfang folgende Schleife steht:

```
1:  do {
2:     c = mygetchar();
3:  } while (c != '*');
```

Listing 1: Die fehlerhafte Version in Sozobon...

Wird ein Kommentar nicht abgeschlossen, bleibt HCC in dieser Schleife hängen, da das Dateiende nicht angeprüft wird. Mit der folgenden Änderung läßt sich der Fehler beheben:

```
1:  do {
2:     if ((c = myget
            char()) == EOF)
3:        fatal("unbalanced
              comment
           delimiters");
4:  } while (c != '*');
```

Listing 2: ...und die verbesserte.

Weitaus schwerwiegender ist ein Fehler im Assembler-Programm JAS (V1.1). Bei den Bitoperationen (BSET, BCLR, BCHG, BTST) mit direkter Angabe der Bit-Nummer wird falscher Objekt-Code erzeugt. Der Assembler setzt die Bit-Nummer in ein 32 Bit breites Feld, anstatt in ein 16 Bit breites. Auch dieser Fehler läßt sich einfach beheben, wenn man den Quelltext besitzt. Dazu müssen nur in der Datei OPCODES.H alle Size-Flags bei den Bit-Operationen von S_L in S_W bzw. von F_L in F_W geändert werden. Wer den Quell-Code nicht besitzt, kann trotzdem ohne weiteres mit dem Compiler arbeiten, da die die Bit-Befehle von HCC nicht verwendet werden (zumindest nicht in Version 1.01).

Andreas Engel, Mainhardt

Red.: Obwohl man Kommentare üblicherweise abschließen sollte, bedanken wir uns für den Tip.



PD-JOURNAL-ABONNEMENT

Bücher / Software / ST Computer - Zeitschriften

erhalten Sie bei Ihrem Computer-Händler und im Buchhandel.

einsenden an:

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt 13 wiederrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Name/Vorname

Straße, Nr.

PLZ, Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM ________ liegt bei.

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberstadt widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Datum Unterschrift

Ja, bitte senden Sie mir das Public Domain Journal ab ________ für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 50.- frei Haus. (Ausland: Nur gegen Scheckvoreinsendung DM 70.- Normalpost, DM 100.- Luftpost). Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Datum/Unterschrift



WAHNSINN !
Das Einsteiger-Paket:
3 Ausgaben STplus
inkl. 3 randvollen Programmdisketten
+ Ordner
zum Superpreis von nur
DM 49,-
Das Sammelwerk mit Diskette für den Atari ST
DM 19.80
ST plus
Ausgabe 1
Ausgabe 2
Ausgabe 3
Workshops
Jetzt einsteigen!
STplus ist das Sammelwerk, durch das
ST User jeden Monat ihren Wissenstand
über alles Wissenswerte rund um den ST
erweitern: DFÜ, CAD, DTP, Emulatoren,
Grafik, Datenbanken, Midi & Musik, Assembler,
Basic, C, Modula 2 und Pascal!
Jeden Monat mit einer
randvollen Programmdiskette!
Ausschneiden und abschicken an: COMPUTEC Verlag, Gufidaunerstr. 13, 8501 Schwarzenbruck
Ich will das STplus Einsteiger-Paket zum Superpreis
von nur DM 49,– (nur gg. Vorkasse – Bargeld od. Scheck!)
Meine Adresse:
VORNAME
NR
WOHNORT

Immer up to date

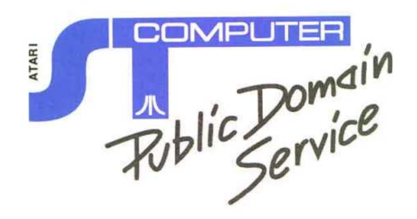
Mit dieser Sparte wollen wir allen unseren Lesern die Möglichkeit geben, sich über die neuesten Programm-Versionen zu informieren. Angegeben werden die aktuelle Versionsnummer, ein eventueller Kopierschutz, die Bildschirmauflösungen und der Speicherbedarf. Softwarefirmen ist es somit möglich, die ST-Computer-Leser über ihre Updates zu informieren.

Programmname	Version	Daten
Adimens ST	3.0	N HM
Adiprog SPC Modula	1.1	N HM
Aditalk ST	3.0	N HM
Adress ST / Check ST	1.0	N H
Afusoft Morse-Tutor	2.0	N HML
Afusoft Radio-Writer	1.0	N HML
Afusoft Radiofax plus	1.0	N HML 1M
AIDA	1.1	N HM
AnsiTerm	1.4	N
Assembler Tutorial	1.06	N HM
Banktransfer	1.0	N H
1st BASIC Tool	1.1	N HML
BTX/VTX-Manager	3.0	N H 1M
Calamus	1.09	N H 1M
Cashflow	1.0	N H 1M
Chips At Work	1.0	N HM
CIS-L&G	1.01	
Creator	1.1	N H
Diskus	1.02	N HM
dBMAN	5.10	N HML
Easytizer	1.0	N HM
Easy Rider Assembler	2.04	N HM
Easy Rider Reassembler	2.31	N HM
fibuMAN	3.0	N H
fibuSTAT	2.3	N H
Flexdisk	1.3	N HML
FM-Meßtechnik	1.0.b	N HM
FTL Modula-2	1.18	N HM
Gadget	1.2.5b	N H
GEMinterface ST	1.1.	N HML
GFA-Artist	1.0	N L
GFA-Assembler	1.2	N HML
GFA-BASIC 68881	1.3	N HML
GFA-BASIC-Compiler	3.5	N HML
GFA-BASIC-Interpreter	3.5	N HML
GFA-Draft plus	3.0	N
GFA-Farb-Konverter	1.2	N H
GFA-Monochrom-Konverter	1.2	N ML
GFA-Objekt	1.2	N HM
GFA-Starter	1.1	N HML
GFA-Vektor	1.0	N
G+Plus	1.4	N HML
GrafStar	1.0	N H
Hänisch Modula-2	3.1	N HML
Hard Disk Accelerator	1.0	N HML
Hard Disk Sentry	1.10	
Hard Disk Toolkit	2.0	N HM
Harddisk Utility	2.2	N HM
Harlekin	1.0	N H 1M
Imagic	1.1	N HML
Intelligent Spooler	1.10	N HML
Interlink ST	1.89	N HM
Junior Prommer	2.33	N HM
K-Resource	2.0	N HM
Kleisterscheibe	2.2	N HM
Label ST	1.0	N HML
Laser C (Megamax)	2.1	N HML
1st_Lektor	1.2	N HM
Lern ST	1.22	N HML
Link_it GFA	1.1	N HML
Link_it Omikron	2.0	N HML
MagicBOX ST	7.75d	N H 1M
Mathlib	3.0	N HM
Mega Paint II	2.30	N H 1M
Mega Paint II Professional	2.31	N H 1M
Megamax Modula 2	3.5	N HM
MGE Grafikkarte	2.0	N
MGP GAL-Prommer	1.03	N H
Micro C-Shell	2.70	N HM
MT C-Shell	1.2	N HM 1M
Multidesk	1.82	N HML
Musix32	1.01	J H
NeoDesk	2.05	N HML
Omikron Assembler	1.86	N HML
Omikron BASIC-Compiler	3.06	N HML
Omikron BASIC 68881-Compiler	3.06	N HML
Omikron BASIC Interpreter	3.03	N HML
Omikron DRAW! 3.0	3.01	N HML
Omikron EasyGEM-Lib	1.0	N HML
Omikron Maskeneditor	1.0	N HML
Omikron Midi-Lib	2.1	N HML
Omikron Numerik-Lib	1.2	N HML
Omikron Statistik-Lib	1.5	N HML
PAM's TERM/4014	3.012e	N H
PAM's TurboDisk	1.7	N HML
PAM's NET	1.1	N HML
PCB-layout	1.19	N H
PC ditto Euroversion	3.96	N HML
PegaFakt	1.3	N H
phs-BTX-Box	6.0	N HML 1M
phs-ST-Box	1.2	N HM
phs-Boxtalk	1.0	N HM 1M
phs-Boxedi	1.0	N HML 1M
phs-Cheapnet	1.2	N HM
Platon	1.45	N H
Prospero Pascal	2.151	N HML
Prospero Fortran	2.152	N HML
Prospero C-Compiler	1.142	N HML
Prospero Developers Toolkit	1.103	N HML
Protos	1.1	N H 1M
Quick_Dialog	1.0	N HM
ReProk	1.10	N H 1M
Revolver	1.1	N HML 1M
Search!	2.0	N HM
Signum! zwei	1.0	N H
Skylink	1.5	N H 1M
Skyplot+	4.1	J H 1M
Soundmachine II	1.0	N HM
SoundMerlin	1.01	N HM
SPC-Modula-2	2.0	N HML
Spectre 128	1.9	J HM
1st_Speeder 2	1.0	N HML 1M
SPS ST	1.5	N H 1M
STAD	1.3+	N H
Steuer-Tax 2.9	1.10	N HM
Steuer-Tax 3.9	1.10	N HM
STop	1.1	N HM
ST Pascal plus	2.08	N HM
SuperScore	1.4	J H 1M
Tempus Editor	2.05	N HM
Theca Librarian	1.0	N HM
TIM	1.2	N H
TIM II	1.0	N H 1M
Transfile ST 1600	1.1	N HM
Transfile ST 850	1.1	N HM
Transfile ST plus	3.0	N HM
Turbo C	1.1	N HM
Turbo ST	1.8	N HML
UIS II + Hermes	2.5	
V_Manager	3.02	N H
VSH Manager	1.0	N HML 1M
WERCS Resource-Editor	1.0	N HM

Irrtum vorbehalten!

Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung, L = niedrige Auflösung, 1M = mindestens 1 Megabyte

Inserentenverzeichnis



Liebe PD-User ,

der Diskussion 'Demos in der ST-Computer PD-Serie' müssen wir uns oft stellen. Unsere Position ist dabei ein sehr klares Nein - keine Demos in der PD-Serie. Wir wollen keine 'Schnupperversionen' kommerzieller Software aufnehmen, da sonst das Niveau der Serie schlagartig absackt, und uns liegt sehr viel daran, dieses hoch zu halten. Uns ist natürlich bewußt, daß die Programmierer für Ihre Arbeit auch belohnt werden wollen, doch dann bitte nicht auf diesem Wege.
Wenn Autoren den Source-Code oder ein gedrucktes Handbuch anbieten bzw. erweiterte Versionen des Programms entwickeln, so ist das in Ordnung, doch wenn ein Programm absichtlich abgespeckt wird, wie Adreßverwaltungen mit 10 Datensätzen oder gesperrter Druckausgabe, Programme mit nur 20% funktionierenden Features oder Spiele die zeitlich begrenzt oder nur einen Teil der Level beherrschen, so ist es für uns eindeutig - solch ein Programm hat in einer PD-Serie nichts zu suchen.
So haben wir Ihnen in diesem Heft beispielsweise die PD-Version eines auf 4 Level begrenzten Spiels 'vorenthalten', welches zwar sehr schön, doch nach etwa 15 Minuten durchgespielt war. Dafür gibt es Demo-Serien. Unsere User sollen sicher sein, daß alle Programme funktionsfähig und dem Anwender nützlich sind. Natürlich sollte dabei nicht vergessen werden, daß Programmierer nur deshalb den Demo-Weg einschlagen, weil es zu wenige Anwender gibt, die sich gegen eine meist sehr geringe Gebühr registrieren lassen, oder dem Programmierer eine Spende zukommen lassen.
Denken Sie 'mal darüber nach.

Ihre ST-Computer Redaktion

DIE NEUHEITEN



MODULA-2 SAMMELDISK 2

HK_LIB 1.0: Sammlung unterschiedlicher Routinen für das LPR Modula-2 (PD 225), u.a. sind Module zur Speicherverwaltung, für XStrings, Longsets und den Timer enthalten.

GRANULE: besseres Speicherverwaltungsmodul für TDI Modula-2.

MODPRINT: Modula-2 Pretty Printer

PLO: der PLO Compiler aus Wirth's Compilerbau Buch.

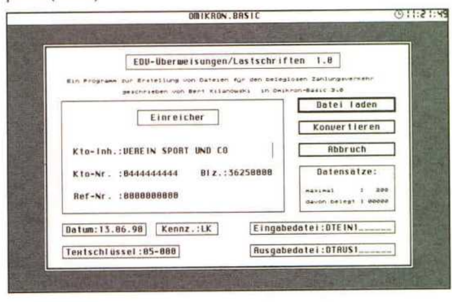


MODULA-2 SAMMELDISK 3

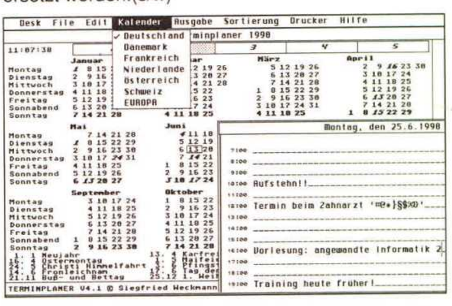
SGLTOOLS: Ein Füllhorn von Utilities zu und für TDI Modula-2, eine vollständige GEM-Library mit Quelltext, erleichtert die Programmierung mit dem TDI Compiler ungemein.

336 VERWALTUNG

GIRO STB ist die neueste Version des Programm von PD 311, da sie erheblich vergrößert, verbessert und erweitert worden ist gibt es nun dieses Programm neu auf PD. Es wurden Fehler des GFA-COMPILERs behoben. GIRO STB unterstützt den beleglosen Datenträgeraustausch mit Banken und Sparkassen. Das Programm wurde den Ansprüchen der Praxis angepaßt (1MB).



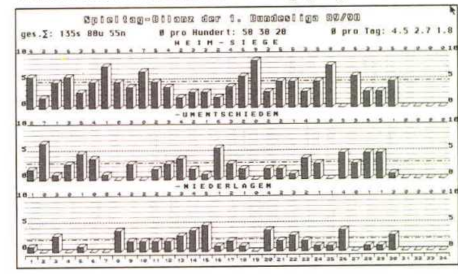
EDV_BUCH ist ebenfalls ein hilfreiches Programm für den beleglosen Datenträgeraustausch. Überweisungen bzw. Lastschriften können damit durch Disketten ersetzt werden.(s/w)



TERMIN V4.0 Termin ist ein Programm das die Terminplanung übernimmt. Es besitzt einen kompletten Jahreskalender auf einer einzigen Maske, wobei dort die Feiertage, nicht nur für Deutschland, markiert. Die Termine werden in den Kalender angezeigt und man kann sich den nächsten Termin einblenden.

337 SPORT-TABELLEN

Verschiedene Tabellenverwaltungsprogramme:



BUTA ST ist ein Verwaltungsprogramm für die 1. und 2. Bundesliga, ideal für Statistiker und Totospieler



P_TAB ist ebenfalls ein Tabellenverwaltungsprogramm für die Bundesliga mit Beispielen aus der Saison 89/90.(s/w)

SPORTTABELL ist ein allgemein gehaltenes Sporttabellenprogramm. Es eignet sich für alle Sportarten, da der Modus frei bestimmt werden kann (s/w)

ABTEILUNG: Dieses Programm unterscheidet sich von den anderen in sofern, daß es keine Tabellen verwaltet sondern die Abteilung.
In der Abteilung können maximal 20 Mannschaften und 30 Spieler pro Mannschaft sein, dies ist aber für den durchschnittlichen Verein mehr als ausreichend.(s/w)
Ligadrei_PD: Ligadrei_pd ist ein Manager und Trainerspiel. Man kann sich eine drittklassige Mannschaft aussuchen und diese dann so trainieren, daß man den Aufstieg schafft, es können Spieler dazu gekauft und verkauft werden (s/w).



UTILITIES

CROSSREF druckt ein formatiertes Cross-Reference-Listing eines GFA-Basic Programms aus.

DHRYSTONE bekannter Referenz-Geschwindigkeitstest.

DISKV3.0 ist ein Disketten-Utility zum Formatieren und Kopieren. Weiterhin führt es einen Viren-Check aus und gibt Informationen über Diskette und Programm. (s/w)

ENVIRON: Dieses Programm dient zum Anlegen von Enviroment-Variablen, damit Programme diese benutzen können.

EXEC_OS mit diesem Programm ist möglich, dem GEM auch ein Enviroment mitzugeben, d.h. einen Suchpfad unter PATH=..., unter dem es dann die Resource-Dateien sucht.

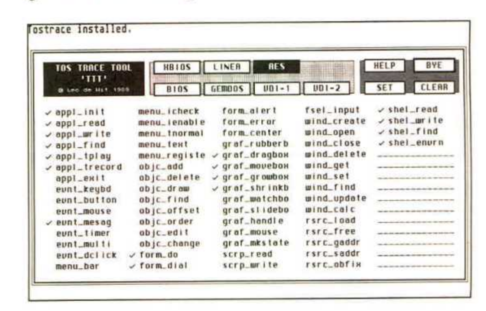
HD_Test sucht defekte Cluster/Sektoren auf der Festplatte und markiert diese. (s/w)

PINHEAD beschleunigt den Bootvergang

POOLFIX3: Korrigiert einen Gemdos-Fehler bezüglich der Speicherverwaltung in TOS 1.4 bzw. 1.6. Der Fehler tritt bei zahlreichen Malloc-Aufrufen auf und in der Speicherverwaltung von Ordnern (ähnlich dem 40 Ordner-Problem)

RENBIN dient dazu, GST Objekt-Code Dateien umzubenennen. Um z.B. Routinen in den Bibliotheken zu ersetzen.

TOPSHELL: Mit diesem Programm kann man seine eigenen Programme in Pull-Down-Menüs hineinschreiben und sie dann durch anklicken starten.
Ideal für Festplattenbesitzer, die verschiedene Programme sehr häufig benutzen.



TOS TRACE TOOL: Das Programm erlaubt das Tracen einzelner Betriebssystemroutinen, egal ob VDI, AES, BIOS, LINE-A, XBIOS und GEMDOS. Sehr praktisch für Systemprogrammierer.

TOSVERS zeigt die Versionsnummern von TOS, GEMDOS und AES an, einschließlich des TOS Datums und die Bezeichnung.

VDISK hierbei handelt es sich um eine residente restfeste RAMdisk, die als Accessory ausgeführt ist.

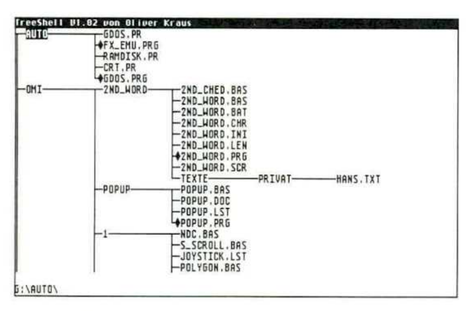
XREFBIN mit Hilfe dieses Programms kann man Bibliotheken im Format des GST Linkers untersuchen, d.h. es liefert eine Liste der in der Bibliothek vorliegenden Module.



CLI-UTILITIES

COMMAND: Komfortable Kommandooberfläche (CLI), ähnlich MS-DOS. Command stellt neben bekannten Kommandos wie 'Dir' 'Format' oder 'Copy' auch Batch-Betrieb und zahlreiche Kontrollstrukturen zur Verfügung. Somit kann beispielsweise auf einfache Weise eine Compilerumgebung geschaffen oder sonstige komplexen Batch-Anweisungen à la MS-DOS erzeugt werden.

CO_CLI: Kopier- und Backup-Programm für CLI. Kopieranweisungen können sich ebenfalls auf ganze Ordner, Wildcards, Archivbits oder Zeitkriterien beziehen. Entgegen normaler Kopierprogramme, wird bei voller Diskette zum Wechseln aufgefordert. Es ist eine ST- und eine DOS-Version enthalten.

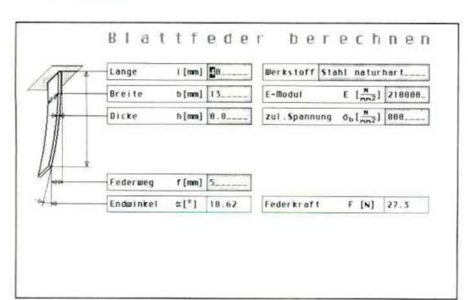


TREESHELL: Komfortables Tool ähnlich XTREE oder QUICKDOS bei MS-DOS. Die Direktorystruktur wird in einer Art Baum angezeigt, wobei diese abgespeichert wird und somit jederzeit schnell vorhanden ist.
In diesem Baum kann man mit dem Cursor auf und ab gehen und erhält somit einen Überblick über alle Dateien und Ordner der Diskette/Platte, bzw. kann die Programme aufrufen.

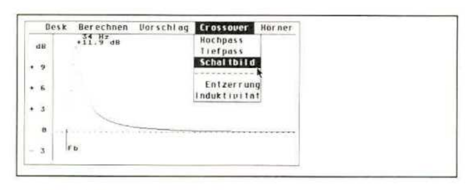
BM_SEARCH: CLI-Tool. Sucht Strings in Dateien.



TECHNIK

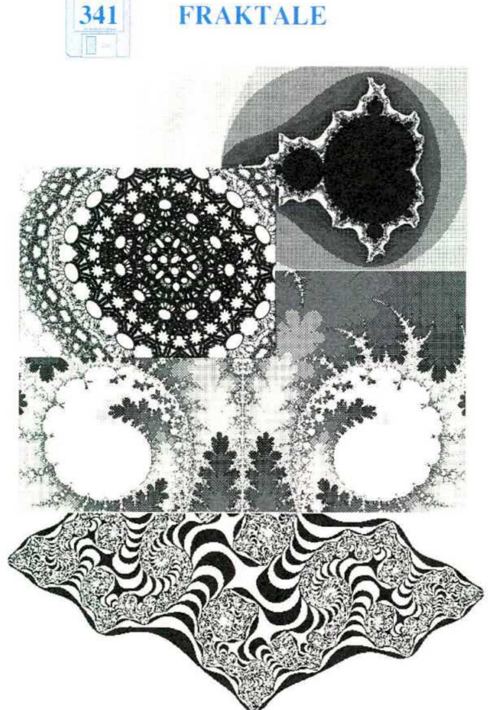


FEDER: Hier sind jede Menge nützliche Programme zusammengefaßt, die zur Berechnung von verschiedenen Federn und Schnappverbindungen dienen. Man kann Zugfedern und Druckfedern (UPDATE PD 332) berechnen, sowie Blattfedern, Schenkelfedern. Es können Schnapphaken und Schnappringe für Schnappverbindungen berechnet werden. Die Teile werden anschaulich beschriftet und maßstabsgetreu gezeichnet. (s/w)



SPEAKER: Ein extrem nützliches Werkzeug zur Konstruktion von Lautsprechergehäusen. Man braucht verschiedene Daten wie z.B. Resonanzfrequenz, Gesamt-Gütefaktor und das Luftvolumen der Box. Das Programm errechnet daraus den Frequenzgang der Box und stellt ihn grafisch dar (wie es in HiFi-Zeitschriften zu sehen ist).

341 FRAKTALE



APFEL: Eine Exkursion ins Land der Apfelmännchen (1MB,s/w).

FATOU: Ein weiteres Kapitel der fraktalen Grafik (s/w).

HÜPFER: Hüpfer ist ein sehr schön gemachtes Programm zur Berechnung von fraktalen Grafiken. Sämtliche Werte lassen sich, wie in den anderen Programmen auch, frei definieren (s/w).

JULIA: Die altbewährten Julia-Mengen dürfen auch nicht fehlen (1MB, s/w)
(s/w)



DRUCKERTREIBER

Auf dieser Diskette befinden sich verschiedene Drucker- und Hardcopytreiber.

BARREL: Druckerspooler. Übernimmt das Drucken - das Warten entfällt.

CFGINSTALL: Konvertiert 1stWordPlus HEX- in CFG-Dateien und umgekehrt.

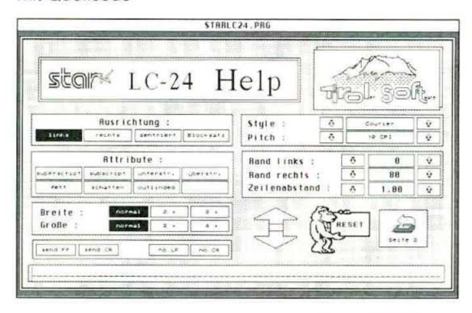
HARDCOPY: Hardcopytreiber für Epson 9 und 24 Nadel-Drucker

HARDCOPYOKI: Hardcopytreiber für OKI-Drucker (s/w)

HPGL-TREIBER: GEM VDI-Treiber für HPGL-Laserdrucker. Der Treiber lenkt, nach zuvorigem Starten von GDOS, die Druckausgabe des ST auf den Plotter um.

HP_DESK: HP-DeskJet Hardcopy Treiber(s/w)

KXP1124: Treiber für Panasonic-Drucker KX-P1124 mit Quellcode



Sowie Druckerteiber für Star LC10, LC24, EPSON LQ500 und NEC P2200(s/w)



DRUCKUTILITIES

CASSETTE: Cassette ist ein gut gemachtes Kassetten-Label-Druckprogramm.
Zur optischen Kontrolle erscheint der spätere Ausdruck noch einmal auf dem Bildschirm (s/w).

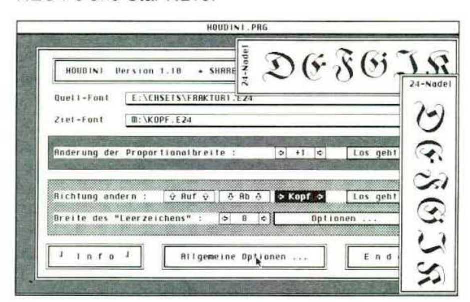


CODES: Kontrollieren verschiedener Codes (u.a. ISBN und Barcode). Das Programm druckt ferner den Barcode in guter Qualität auf dem Drucker aus oder zeigt ihn am Bildschirm (s/w).

DIRECTOR: Director erstellt ein übersichtliches Diskettenverzeichnis, wobei verschiedene Auswahlmasken möglich sind. Der Druck auf 3.5 Zoll-Etiketten ist eingebaut.

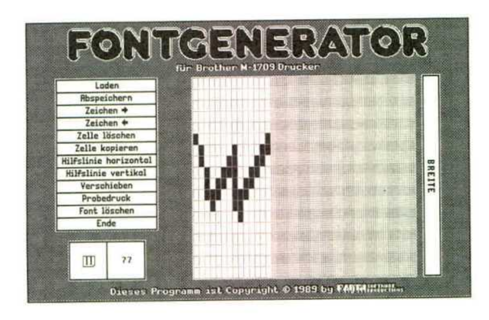
DRUCKLST: Mit diesem Programm ist möglich, ein formatiertes Listing eines mit GFA-BASIC erstellten *.LST Files zu drucken.

GITTER: Das Programm Gitter ist Download-Zeichensatzeditor, der in zwei Versionen vorhanden ist, für NEC P6 und Star NL10.



HOUDINI: seines Zeichens Entfesslungskünstler, entfesselt hier SIGNUM!, denn es ermöglicht, Signumfonts um 90° zu drehen und als neuen Font abzuspeichern. 90° 180°und 270° sind möglich. Das schafft neue Möglichkeiten in SIGNUM!, z.B. seitliche Beschriftungen, neue Briefköpfe,...

ST-COMPUTER PUBLIC DOMAIN



FONTGEN: Fontgenerator für Brother M-1709 Drukker, dieser Fontgenerator dient zur Erzeugung von Download-Fonts. Da der Brother M-1709, laut Hersteller, kompatibel zu Epson-Drucker ist, sollte es eigentlich keine Probleme mit diesem Programm geben.(s/w)

LABYRINTH: Erzeugt und druckt ein Labyrinth.

MESSMETA: Messmeta ist ein Programm zum Ausdrucken von Messdaten des Programms Messwert (PD 57) unter GDOS, d.h. in höchster Druckqualität. Die Größe kann man hierbei von DIN A4 bis Briefmarkengröße varieren.

PRNOUT: HARDCOPY für NEC P2200. Besonders interessant für die Drucker der Serie mit vereinzelten Übertragungsschwierigkeiten (kleine Fehler im Ausdruck). Diese kann man mittels eines Interfaces oder mittels dieses Programms beheben.

ZPRINT: ZPRINT dient als Druckprogramm für beliebige ASCII-Dateien (s/w).

344 KOMPRIMIERUNG



Auf dieser Diskette befinden sich Programme zum Komprimieren von Daten, Programmen oder ganzen Ordnern. In der DFÜ-Welt gerne gesehen, doch auch für 'unverkabelte' ST'ler sehr praktisch, lassen sich doch bis zu 70% Platz einsparen.

LHARC: Die neueste Version von LHarc 1.13 wurde optimiert und besitzt nun die Versionsnummer 1.1309. Außerdem findet sich auf der Diskette der Quelltext von LHarc - eine Hilfe für alle, die eigene Komprimierungsroutinen schreiben wollen.

ARC 6.02: Das bekannte Archivierungsprogramm packt nun auch ganze Ordner. Weiterhin ist eine Shell vorhanden, die das Ordner-Packen unterstützt, aber auch eine komfortable Shell für ältere ARC-Versionen (PD 263) kommt nicht zu kurz.

ZOO 2.1: Eines der meistbenutzten Archivierungsprogramme auf dem ST - die neueste Version.



KOMPRIMIERUNG

UNZIP: Endlich ist es soweit! Auch ZIPs, die von einem PC gepackt wurden, können bearbeitet werden. Es ist voll kompatibel zu allen auf MS-DOS-Rechner gepackten Dateien.

ZOO-Source: Programmierer und Interessierte können sich auf den Quelltext von ZOO freuen. Er ist in C geschrieben und enthält viele Anmerkungen.

ABKÜRZUNGEN

1MB = mind. 1MB Speicher notwendig
s/w = nur Monochrom; f = nur Farbe

UPDATES

Da die Programmierer unserer PD-Programme sich nicht schlafen legen oder sich auf Ihren Lorbeeren ausruhen gibt es ständig Erweiterungen oder Verbesserungen der Programme. Diese Verbesserungen beruhen auch teilweise auf Ihren Anregungen oder Wünschen, denn das beste Programm ist ohne den User fast nichts wert.

PD 115 Kreuzworträtsel: man kann nun die Fragen und Antworten in ASCII-Format speichern, ferner ist es erheblich schneller geworden und hat jede Menge kleine Verbesserungen bekommen.
PD 147 Datei100: fehlerbereinigt
PD 244 IdeaList: Druckerprobleme in Verbindung mit Nano-Schriftgröße wurden behoben
PD 245 SET: läuft jetzt in allen Auflösungen
PD 248 Linguix: Fehler bei der Wordplus-Doc-Seitenzählung wurden behoben und um viele Funktionen erweitert
PD 267 Galaxie: verfeinert
PD 299 WPlusTeX: Konvertiert Wordplus-Texte nach TeX (PD 250-254, 276,277,278) Grafikeinbindung wurde erheblich verbessert.
PD 305 ADRETI ist wesentlich schneller und um einige Funktionen erweitert worden
PD 308 VAULT III: stark überarbeitet und Fehler wurden beseitigt, PD 311 SALDI: Grafikteil ist neu dazugekommen und Fehler wurden beseitigt
PD 312 U110: Benutzerführung wurde verbessert und Fehler beim Laden der zweiten Bank wurde elemeniert
PD 317 FUNKTIONSPLOT überarbeitete Version, Fehler wurden ausgemerzt
PD 317 TABLO Fehler wurden beseitigt
PD 320 CLIPBOARD neue Version
PD 330 STARFINDER ist um eine weitere Objektdatei erweitert worden
PD 331 Bio_rtm380: jetzt mit Partnervergleich!

MACHEN SIE MIT!

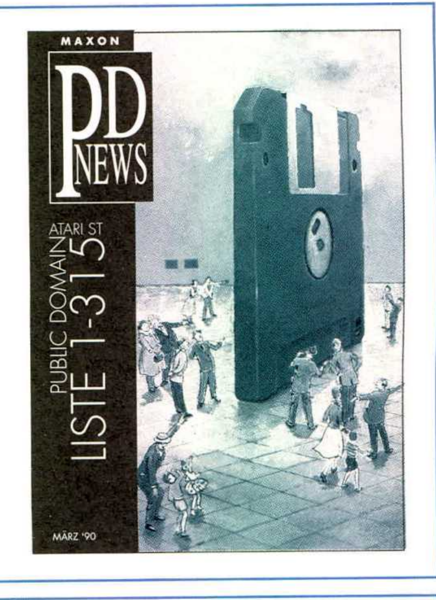
Möchten Sie ein selbstgeschriebenes Programm in unsere PD-Sammlung geben, um es auch anderen Usern zugänglich zu machen? Kein Problem. Schicken Sie es uns auf einer Diskette zu, samt einer Bestätigung, daß es von Ihnen geschrieben wurde und frei von Rechten Dritter ist. Bei Fragen steht Ihnen die Redaktion gerne zur Verfügung.

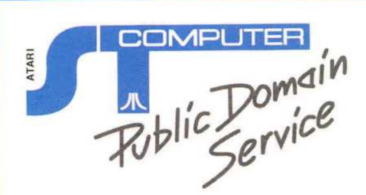
MAXON Computer • ST-Computer PD
Industriestr. 26 • D-6236 Eschborn

PD EN MASSE!

Wegen der großen Nachfrage haben wir sie ins Leben gerufen - die PD-News. Sie enthält die komplette Übersicht aller ST Computer-PD-Disketten. Sie erhalten sie - wie gewohnt - kostenlos gegen Zusendung eines mit DM 2.40 frankierten und an Sie adressierten *DIN A4*-Rückumschlages, denn in kleinere paßt unsere Zeitschrift beim besten Willen nicht rein!!!

MAXON-Computer • PD-NEWS
Industriestr. 26 • D-6236 Eschborn





DIREKT-VERSAND

Alle PD-Disketten unserer Sammlung gibt es direkt bei MAXON-Computer.

Um einen schnellen Versand zu ermöglichen, beachten Sie bitte folgende Punkte:

1. Schriftliche Bestellung

- Der Unkostenbeitrag für eine Diskette beträgt DM 10,-
- Hinzu kommen Versandkosten von DM 5,- (Ausland DM 10.-)
- Bezahlung per Scheck oder Nachnahme
- (Im Ausland nur Vorrauskasse möglich)
- Ab 5 Disketten entfallen die Versandkosten (DM 5.- bzw. DM 10.-)
- Bei Nachnahme zuzüglich DM 4,00 Nachnahmegebühr

2. Telefonische Bestellung

MAXON-Computer GmbH
'PD-Versand'
Tel.: 0 61 96 / 48 18 11
Fax: 0 61 96 / 4 18 85

Mo-Fr 9⁰⁰ - 13⁰⁰ und 14⁰⁰ -17⁰⁰ Uhr

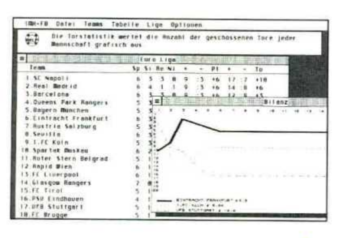
- Lieferung erfolgt per Nachnahme

Adresse:

MAXON-Computer GmbH
'PD ST-Computer'
Postfach 5969

Nutzen Sie die PD-Karte in diesem Heft





FUSSBALL NEU

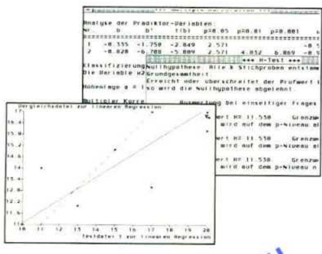
Tabellenverwaltung

Rechtzeitig zur neuen Saison! FUSSBALL dient zur Verwaltung und Auswertung spezieller Informationen für Fußball-Fans, d.h. Tabellenstände, Torschützen, rote/gelbe Karten oder Besucherzahlen. Das Programm erzeugt nach Eingabe der Ergebnisse die aktuelle Tabelle, und wertet die Daten gezielt statistisch aus.

Dazu gehört u.a.: Mannschaftsstatistik: z.B. Stadionauslastung, Besucherzahl und -schnitt, Rekordbesuch, bester Torschütze, Gesamtzahl gelbe/rote Karten..., Rundenstatistik: dto für gesamte Runde bzw. Saison, Team-bezogene Höchst- bzw. Tiefstwerte.

FUSSBALL beinhaltet die kompletten Daten der Saison 89/90 (östreichische und Bundesliga), sowie die Paarungen der neuen Saison 90/91. Da sich das Punktesystem frei definieren läßt (z.B. auch 3 Punkte bei Sieg) lassen sich damit auch andere Sportarten verwalten.

FUSSBALL[1]
SD 34 DM 15.-



STatiST NEU

modulares Statistik-Programmpaket

STatiST ist ein umfangreiches Paket zur Auswertung statistischer Daten.
Es beinhaltet folgende 35 Testverfahren:

Bartlett-Test • Basis-Statistik • Box-Test • chi^2-Test nach Brandt und Snedecor • Dreifaktorielle Kovarianzanalyse [mit Meßwiederholung] • Dreifaktorielle Varianzanalyse [mit Meßwiederholung] • Einfaktorielle Diskriminanzanalyse • Einfaktorielle Kovarianzanalyse [mit Meßwiederholung] • Einfaktorielle Varianzanalyse [mit Meßwiederholung] • F-Test • Friedmans Rangvarianz • H-Test • Hartley-Test • Kendalls tau • Kolmogorov-Smirnov-Test • Life-Table (Überlebenskurven) mit Logrank-Test • Lineare Regression • Multiple Korrelation • Odds Ratio • Partial-Korrelation • Scheffe-Test • Spearmans Rangkorrelation • t-Test für abhängige Stichproben • t-Test für unabhängige Stichproben • U-Test • Wilcoxon-Paartest • Wilcox-Wilcoxon-Test • Zweifaktorielle Kovarianzanalyse [mit Meßwiederholung] • Zweifaktorielle Varianzanalyse [mit Meßwiederholung]

Zu jedem Prüfverfahren werden sämtliche Ergebnisse mit dem entsprechenden Wertungen und Kommentaren ausgegeben und, falls möglich, grafisch angezeigt.
STatiST eignet sich für sämtliche, z.B. im Studium erforderlichen statistischen Auswertungen und macht das zeitaufwendige Rechnen per Hand und das Arbeiten mit Tabellen überflüssig.

STATIST[1] (2 Disketten)
SD 32a/b DM 30.-



EASYSTAT NEU

Induktive Statistik

EASYSTAT dient der Errechnung und Veranschaulichung statistischer Verfahren. Dabei wird neben der beschreibenden Statistik vor allem die induktive Statistik berücksichtigt.
Es eignet sich für alle Anwender der Statistik (Wirtschafts- und Sozialwissenschaftler, Techniker und Studenten).
Mit EASYSTAT können Daten eingegeben, dargestellt, Kennzahlen berechnet, nach Zusammenhängen gesucht und Tests bzw. Intervallschätzungen durchgeführt werden. Von folgenden zentralen Verteilungen können Quantile (oder Pseudoquantile), Verteilungsfunktion und Wahrscheinlichkeitsfunktion (bzw. Dichte) berechnet werden: Geometrische Verteilung, Hypergeometrische Verteilung, [Negative] Binomialverteilung, Poissonverteilung, Normalverteilung und Standardnormalverteilung, t-Verteilung, Chi-Quadrat-Verteilung, F-Verteilung, Exponentialverteilung, Erlang-n-Verteilung, Weibullverteilung, Beta-Verteilung, Gamma-Verteilung.
EASYSTAT soll nicht zuletzt sehr abstrakte Dinge (z.B. statistische Tests) veranschaulichen helfen. Eine eingebaute einfache Kommandosprache ermöglicht es, Testprozeduren selbst zu schreiben.

EASYSTAT läuft unter einer luxuriösen Benutzeroberfläche, beinhaltet einen speziell zugeschnittenen Editor und stellt ein On-Line-Hilfe-System zur Verfügung.

EASYSTAT[1]
SD 31 DM 25.-

Ultra-Disk NEU

RAM-Disk-Tool





Ultradisk ist eine ultraschnelle, größenveränderbare, resetfeste und resetresidente RAM-Disk. Die Größe und die Laufwerkskennung kann frei bestimmt werden, und das alles ohne Inhaltsverlust und ohne den Rechner neu zu booten. Weiterhin enthalten ist ein ultraschneller Drucker-Spooler, der dafür sorgt, daß Sie weiterarbeiten können, während der Rechner noch Daten an den Drucker schickt. Auch darf der Maus-Speeder, die Zeitanzeige und der Bildschirmschoner nicht fehlen.

ULTRA-DISK
SD 33 DM 15.-

ASSOZIATIX

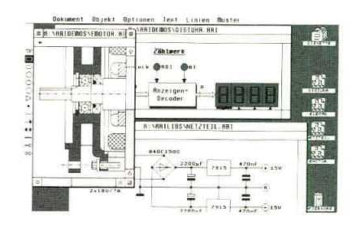
Assoziative Datenbank

Assoziatix ist eine assoziativ-Muster orientierte Datenverwaltung, die es ermöglicht aus einer großen Datenmenge bestimmte Gruppen auszufiltern und daraus dank schneller assoziativer Suche nach bestimmten Zusammenhänge zu finden (z.B. Rasterfandung).
Mit Hilfe des Formulareditors können die Eingabemasken leicht am Bildschirm gestaltet werden, sogar mit Grafikeinbindung.

Einige Besonderheiten:
- Passwortschutz, Export- und Importfunktion, Serienbriefe, Reportdokumentation Statistische Berechnung numerischer Werte Expertfunktion, Volltextsuche Grafikeditor: Spiegeln, Drehen, Zoomen. Balken- Linien und Kuchengrafik.

Mögliche Anwendungen wäre z.B. die Archivierung von Fotomaterial nach Merkmalen, chemische Verbindungen und Eigenschaften von Stoffen bis hin zur eigenen kleinen 'Volkszählung', sei es auch nur zum Austesten was man mit solchen 'persönlichen' Daten so alles machen kann.

ASSOZIATIX (2 Disketten)
SD 27 a/b DM 30.-



ARIADNE

ARIADNE ist ein objektorientiertes Zeichenprogramm, d.h. Objekte können auch im Nachhinein ohne Auslösungsverlust verändert werden (Kreis bleibt Kreis!). Es bietet die Möglichkeit, jedes beliebige Grafikobjekt (mit Doppelklick) zu öffnen, worauf eine neue Zeichenebene bereitgestellt wird. Die Objekte auf dieser Ebene können dann wiederum geöffnet werden usw. Diese hierarchische Struktur eignet sich besonders zur Darstellung komplizierterer Dinge, z.B. Blockschaltbilder, Schaltungen, Software-Dokumentation.

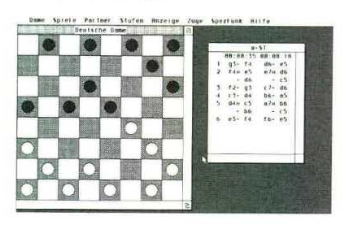
ARIADNE[1]
SD 8 DM 15,-



MASTER ETIKETT

Leistungsstarkes Programm zum Bedrucken von Disketten, Video- und Audio-Cassetten, sowie Paket- und Ordneraufklebern. Bei Diskettenaufklebern können die Daten sogar bequem von der jeweiligen Diskette gelesen und per Klick in den Aufkleber übertragen werden. Mehrere Fonts und Grafikeinbindung unterstützen eine originelle Gestaltung.

MASTER ETIKETT[1]
SD 28 DM 15.-



DAME

Computerumsetzung des alten Brettspiels, wobei der ST einen spielstarken Gegner darstellt. Die Figuren werden per Maus angewählt, die Züge protokolliert und analysiert. Verschiedene Spielstärken, Zugvorschläge, Laden und Speichern einer Partie, sowie verschiedene Spielvarianten dürfen nicht fehlen.

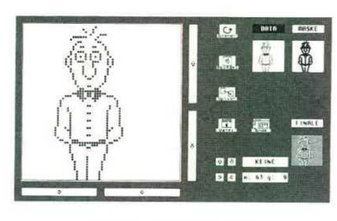
DAME
SD 29 DM 15.-

DER MOTOR

Autotechnik

Der Motor erklärt mit zahlreichen Grafiken die Funktionsweise eines Verbrennungsmotors. Sehr anschaulich sind die bewegten Grafiken, die z.B. die Bewegung eines Kolbens und die Zündzeitpunkte deutlich machen. Das gezeigte Wissen wird zusätzlich in einem Quiz abgefragt. Jetzt mit **geregeltem Katalysator**!!

DER MOTOR[1]
SD 20 DM 15.-



ICONDESIGN

Dieser Icon-Editor ermöglicht das komfortable Erstellen von Icons beliebiger Größe. Grafische Grundfunktionen (Linien, Kreis,...) im Icon-Raster, automatische Maskierung und Erzeugung von Icon-Daten und Source-Code zum Einbinden in eigene Programme, Icon-Ausgabe als RSC-Datei.

ICONDESIGN
SD 17 DM 15,-

SONDERDISK

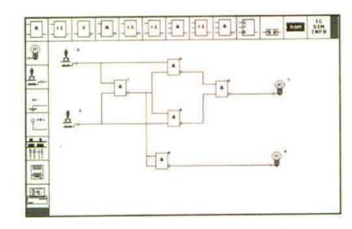


Special Paint 2

Grafik deluxe

Grafikprogramm der Extraklasse. Neben den vielen nützlichen Funktionen zeichnet sich Special Paint vor allem durch seine Geschwindigkeit, seine bequeme Bedienung und seine Kompatibilität zu bekannten Malprogrammen aus. Special Paint bietet umfangreiche Blockfunktionen, Lasso, superschnelle Lupe, Maskierungen, Clippen, schnelle Bieg-, Zerr- und Drehoptionen, Animation und vieles mehr. Clipboardunterstützung, umfangreiche Textfunktionen (ladbare Fonts, Blocksatz, Zeilenumbruch).

Special Paint [1]
SD 21 DM 20.-



ICSIM

Logik-Simulator

Das Programm simuliert das Verhalten von logischen Schaltungen. Bausteine und Verbindungen werden frei per Maus positioniert bzw. verbunden. Eine Schaltung läßt sich somit leicht austüfteln, testen und erst dann in die Praxis umsetzen.
Es sind die Logikbausteine nach DIN 40900 enthalten: AND, OR, NOT, NAND, NOR, XOR, RS-FF, KLEMME, LAMPE, SCHALTER, 0V und +5V.
Die Simulation wird als Impulsdiagramm oder Logiktabelle ausgegeben. Weiterhin liefert das Programm den Schaltplan und eine Liste der benötigten Bauteile.

ICSIM (1)
SD 25 DM 20.-

WÜRFELPOKER

Da sind die Würfelzocker gefragt. Würfelpoker kann man mit mehreren Personen (auch Computer) spielen. Sinn ist es, seine Würfe möglichst geschickt zu verteilen und gezielt nachzuwürfeln. Neben vielen Pärchen gilt es Straßen oder sonstige Kombinationen zu erzielen.

WÜRFELPOKER
SD 30 DM 15.-
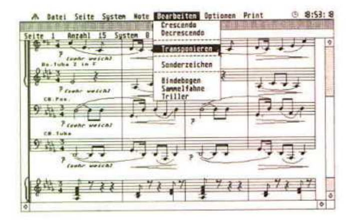
➪ Sonderdisk-Bestellung

Sonderdisks können Sie telefonisch oder schriftlich bestellen, oder nutzen Sie einfach die Bestellkarte im Heft.

Bei Nachnahme zzgl. DM 4.- Gebühr, Versandkosten DM 5.- (Ausland DM 10.-)

MAXON Computer
Industriestr. 26
6236 Eschborn
Tel: 06196/481811

Prospekt gegen frankierten Rückumschlag

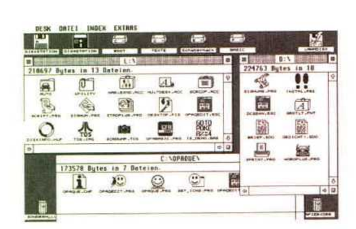


TRISTAN

Notensatzsystem

Für alle Musikfreunde, die nicht nur vom Blatt spielen, sondern auch aufs Blatt schreiben, bietet das Notensatzsystem TRISTAN die ideale Möglichkeit, ihre Noten professionell zu Papier zu bringen. Es lassen sich Partituren mit bis zu 100 Seiten mit max. 32 Notensystemen je Seite bearbeiten. Alle im klassischen Notensatz gebräuchlichen Zeichen, u.a. auch Schlagzeug- und Vorschlagnoten, lassen sich bequem mit der Maus edieren. Ebenfalls stehen mehrere Notenschlüssel, Sammelfahnen, Triller und Bindebögen zur Verfügung. Automatische Transponierfunktion. Ausdruck auf 9- und 24-Nadeldruckern in *maximaler* Druckerauflösung.

TRISTAN
SD 24 DM 25.-



OPAQUE

Das Desktop mit neuem Gesicht

Wie wäre es mit einem zweckmäßigen und originellen Desktop? Opaque bietet die Möglichkeit, jedem Programm ein eigenes, sinnbezogenes Icon zuzuordnen. Auch die Laufwerke lassen sich ändern. Weiterhin kann man die Icons mit Wildcards definieren, z.B. *.BAS, *TXT, *.ACC oder gar WORD????.PRG für verschiedene Programmversionen. Samt Iconeditor und über 100 Icons.

OPAQUE
SD 22 DM 15.-

HAUSKASSE

Kostenplanung

HAUSKASSE verwaltet die Ausgaben eines privaten Haushaltes. Man bekommt dadurch einen Überblick über die angefallenen Kosten und die verfügbaren Mittel, bzw. an welcher Stelle man sparen sollte. Vorausplanung, Vergleich mit tatsächlichen Ausgaben, grafische Darstellung, Fixkostenverwaltung.
Ein ideales Hilfsmittel, um die privaten Kosten in Griff zu bekommen.

HAUSKASSE [1]
SD 26 DM 15.-

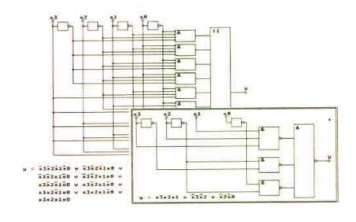
[1] nur für monochromen Monitor (SM 124)
[2] nur für Farbmonitor

HARDCOPY II

Die erste Farb-Hardcopy für den ST

Universelles Hardcopy-Tool. S/W- und Farb-Hardcopy auf *allen* Druckern in *allen* Größen, Screendump auf Disk, Formatkonvertierung, läuft als Accessory, einfachste Bedienung, optimale Druckqualität.

HARDCOPY II
SD 15 DM 15.-



Quinemac

Der Schaltfunktionen-Analyzer

Quinemac optimiert logische Schaltungen Sie geben die Anzahl Ihrer Eingänge und den Zustand der Ausgänge an - und Quinemac liefert die dazugehörige Schaltfunktion (natürlich optimiert), einen Schaltplan und dessen Stückliste.

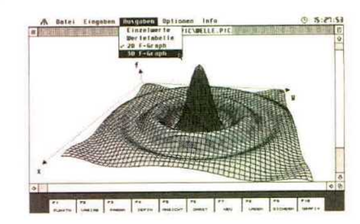
Quinemac [1]
SD 10 DM 15,-

PANDA

Der Farbemulator

Der Farbemulator simuliert die Farbauflösungen des ST auf einem monochromen Monitor (SM, 124,..). Dadurch kann man auch Farbspiele oder sonstige Farbprogramme laufen lassen, die sonst einen zweiten Monitor erfordern.

PANDA
SD 18 DM 15,-



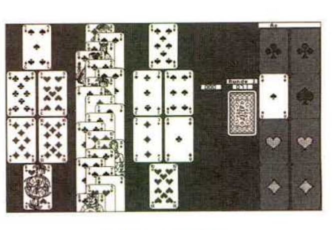
FORMULA

2D-/ 3D-Plotter

2D-/3D-Plotter für mathematisch-wissenschaftliche Anwendung. Der eingebaute Formel-Interpreter beherrscht neben allen gängigen Operationen auch die Definition verschiedener Formeln in bestimmten Teilbereichen, logische Operationen und IF..THEN..ELSE-Kommandos. Somit kann man auch komplizierteste Terme darstellen. 3D-Grafiken lassen sich aus verschiedenen Blickrichtungen anzeigen und mit Schattierungen versehen, was den räumlichen Eindruck verstärkt.

FORMULA
SD 23 DM 20.-

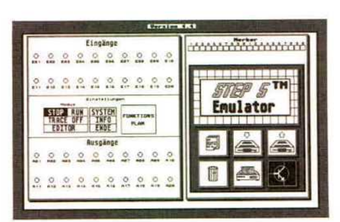
Sonderdisks unterliegen trotz des niedrigen Preises einem Copyright.



PATIENCE

Kartenspiel, das hohe Aufmerksamkeit erfordert. Es schult das Denkvermögen, fördert die Kombinationsfähigkeit, entspannt und beruhigt zugleich. Im Programm sind folgende Patience-Varianten enthalten: Standard, Eiffelturm, Zopf, Kleine Harfe, Matriarchat und Bildergalerie. Das Programm gibt auf Wunsch Lösungsvorschläge. Eine sehr gute Grafik und ein durchdachtes Konzept lassen jedem Kartenfreund das Herz höher schlagen.

PATIENCE [1]
SD 11 DM 15,-



SPS-Emulator

für programmierbare Steuerungen

Unser SPS-Emulator baut auf einem SIEMENS PG 605-Programmiergerät in STEP 5 auf. Mit ihm lassen sich SPS-Programme schreiben, auf Simulationsbasis austesten, laden, speichern, ändern, ausdrucken und als FUP (Funktionsplan mit logischen Gattern) ausgeben. Enthalten sind ein Editor, ein Interpreter und FUP-Generator.

SPS-Emulator [1]
SD 14 DM 15,-

Weitere Sonderdisks im Überblick

01 TOS 1.0	15,-
02 RCS 1.4	15,-
03 Extended VT52[1]	15,-
04 Lovely Helper	15,-
05 Accessories	15,-
06 NIKI[1]	15,-
07 Virus Ex	15,-
09 Legende[2]	15,-
12 MagicBox ST	15,-
13 Robotwar[1]	15,-
16 Easy Adress[1]	15,-
19 MAKI[1]	15,-

Diese Disketten können Sie natürlich auch weiterhin bestellen oder fordern Sie einfach unsere Info an.

- Sonderdisk, was ist das? -

Sonderdisks beinhalten Programme aus den verschiedensten Bereichen (z.B. Utilities, Grafik, Schulung, Spiele). Sonderdisks ermöglichen den Usern, qualitativ hochwertige Software zu einem kostengünstigen Preis zu erhalten. Im Preis ist eine Beteiligung der Autoren enthalten. Haben auch Sie ein Programm für diese Serie, so schreiben Sie uns.

MAXON Computer, Industriestr. 26
'Idee Sonderdisk', 6236 Eschborn

Jetzt im



MAGAZIN PLUS SOFTWARE FÜR DEN ATARI ST & TT

Neue Software Im Test:
Grafikprogramm That's Pixel • Synonymlexikon Lexothek • Sample-Programm Sound Merlin

Programmiersprachen
Test: 4 Modula-Compiler • Tips & Tricks: Omikron-Basic • Empfehlung: Die besten Public-Domain-Sprachen

Drucker Test: Laser mit mehr Funktionen um 500 DM billiger • 8 Nadeldrucker im Vergleich

In der PC-Speed-Fabrik
Hans-Jörg Sack: „Meine erste Million durch den MS-DOS-Emulator"

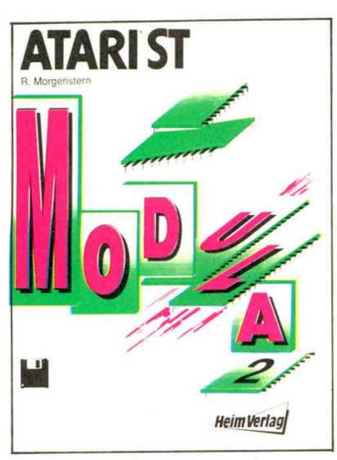


Buchhaltung TIM
Spezialversion zum Praxistest, Vollversion für 50 DM weniger • Flugsimulator • Demo: Back to the future II • Gimmick • Seite 94

Handel

Internationale Computer–Presse
Wendelsteinstr. 3, 8011 Vaterstetten
Tel. 08106/33954, Fax 08106/34238

Buch-Neuheiten



MODULA-2, als konsequente Weiterentwicklung von Pascal und eine der modernsten Programmiersprachen überhaupt nimmt Sie bei Ihren ersten Schritten an die sichere Hand. Begriffe wie Datentypen (Integer, Byte, Pointer, Address...), Datenstrukturen (Felder, Verbunde, Listen), Schleifen (For, Repeat, While, Loop - nein, keine Endlosschleifen), Prozeduren, Prozedurvariablen, Module (lokale, Definitions-, Implementations-, Programm-) und Coroutinen (für parallele Prozesse) werden Ihnen schon bald vertraut sein. Sie steigen voll ins Programmieren ein, und die einzelnen Elemente werden an konkreten Beispielen erklärt, gerade wie sie benötigt und verwendet werden.

Wichtige Merkmale:

- eine komplette Adressverwaltung
- ein UPN-Rechner (nicht 2•3 sondern 23•)
- für diejenigen die lieber 2•sin(x) statt 2 x sin • schreiben, ein Infix-Postfix-Konverter (natürlich inklusive cos, tan, ln, usw.)
- eine Grafikbibliothek (freiwählbares Koordinatensystem, schnelle Linien, Clipping, rotierende Ellipsen, Turtlegrafik
- dynamische Strings (Basic erblaßt vor Neid)
- ein Editor (handlich, klein, fein)
- und im großen Finale ein UPN-Interpreter mit Schleifen, Variablen, Prozeduren und allen Funktionen der Grafiklib, (damit sie Ihre Grafiken interaktiv entwickeln können).

Best.-Nr. B-446
Hardcover

59,- DM



Das Buch fängt wirklich bei Null an und führt Sie Schritt für Schritt in die Programmiersprache ein. Die Beispiele, anhand derer die Sprache, ihre Befehle und Strukturen demonstriert und erläutert werden, wurden aus folgenden Gründen gewählt: Sie sollten auch für Nichtmathematiker nachvollziehbar sein; sie gehören zur täglichen Praxis; sie lassen die besonderen Möglichkeiten des Computers auch für nichtmathematische Bereiche der EDV erkennen; sie verdeutlichen die besonderen Möglichkeiten der wohlstrukturierten Sprache PASCAL.

Der Leser wird vertraut mit den wichtigsten Programmstrukturen, den Möglichkeiten, die Sprache gewissermaßen zu erweitern durch Unterprogramme, denen Parameter übergeben werden können. Alle wesentlichen Grundlagen werden ausführlich erläutert:

- Konstanten, Datentypen und Variable
- Kontrollstrukturen, Bedingungen und Verzweigungen
- Schleifenkonstrukte
- komplexe Datenstrukturen
- Sortier-Algorhythmen, binäres Suchen
- Zeiger, Listen, Bäume, dynamische Datenstrukturen.

Das Buch orientiert sich zwar am für den ATARI konzipierten ST-PASCAL PLUS, berücksichtigt aber auch UCSD-PASCAL und TURBO-PASCAL. Die fast nebenbei zu erarbeitenden Programme sind so gestaltet, daß sie praktisch einsetzbar sind und entsprechend den konkreten Bedürfnissen leicht modifiziert und problemlos erweitert und ergänzt werden können.

Best.-Nr. B-447
Hardcover

49,- DM



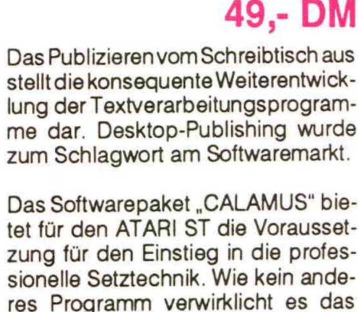
Wesentlicher Inhalt dieses Buches ist die organisatorische und konzeptionelle Typisierung der Anwendungen und der Anwender der EDV im Büro. Dadurch wird eine weitgehende Unabhängigkeit von den laufenden technischen Veränderungen, wie neuer Soft- und Hardware, erreicht. Losgelöst von irgenwelcher „neuesten Soft- und/oder Hardware" wird aus rein organisatorischer Sicht (ausgehend von etwas einleitender Theorie) der Schwerpunkt auf den eigentlichen Kern der Büro-EDV gelegt:

- auf das Kommunikations- und Arbeitsverhalten im Büro
- auf die Anwendertypen und die Anwendungsprofile im Büro
- auf die für die jeweiligen Anwendertypen erforderliche EDV-Unterstützung.

Die EDV-organisatorische und konzeptionelle Betrachtung von Vorgesetzten, Sachbearbeitern, Sekretariat und Büro-Hilfskräften dient in erster Linie der Ermittlung der notwendigen Software-Typen zur optimalen Erledigung der täglichen Aufgaben im Büro. Neu ist auch, daß nicht nur die Vorteile der Büro-EDV dargestellt werden, sondern auch die Nachteile und die Risiken der totalen EDV im Büro, sowohl für den Anwender wie auch für das verantwortliche Management. Wer sich ein Bild über EDV-organisatorische und konzeptionelle Betrachtungsweisen der Büro-EDV machen will, wird mit diesem Buch genauso zufrieden sein wie der Praktiker, der eine Büro-EDV organisieren will oder soll.

Best.-Nr. B-443
Hardcover

49,- DM



Das Publizieren vom Schreibtisch aus stellt die konsequente Weiterentwicklung der Textverarbeitungsprogramme dar. Desktop-Publishing wurde zum Schlagwort am Softwaremarkt.

Das Softwarepaket „CALAMUS" bietet für den ATARI ST die Voraussetzung für den Einstieg in die professionelle Setztechnik. Wie kein anderes Programm verwirklicht es das „what you see is what you get"-Prinzip. Die Druckqualität stellt eine Herausforderung an die Ausgabe-einheit dar. Die kreative Arbeit wird nur zum Erfolg, wenn der Anwender sinnvolle Vorgaben in das Programm einbringt. Ein umfangreiches Vorwissen ist nötig, um mit dem Programm arbeiten zu können. Dieses Buch ist ein unentbehrliches Hilfsmittel für alle EINSTEIGER, UMSTEIGER und AUFSTEIGER in das weite Feld des Desktop-Publishing. In den einzelnen Kapiteln werden Sie mit den Möglichkeiten und den Problematiken der elektronischen Erstellung von Druckvorlagen vertraut gemacht. Viele praxiserprobte Tips erleichtern Ihnen den täglichen Umgang mit dem Softwarepaket „Calamus". Wenn Sie bisher mit „Calamus" keine befriedigenden Ergebnisse erzielen konnten, sollten Sie nicht länger auf dieses Buch verzichten.

Best.-Nr. B-442
Hardcover

49,- DM

Alle Preise sind unverbindlich empfohlene Verkaufspreise

Heim Verlag

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Schweiz
Data Trade AG
Landstr. 1
CH - 5415 Rieden - Baden

Österreich
Haider
Computer + Peripherie
Grazer Str. 63
A - 2700 Wiener Neustadt

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir: ____________________

zuzügl. Versandkosten DM 6,– (Ausland DM 10,–) unabhängig von der bestellten Stückzahl

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

In der nächsten ST-Computer lesen Sie unter anderem

Bildverarbeitung mit dem ST

In der nächsten Ausgabe der ST-Computer wollen wir Ihnen das Thema Bildverarbeitung mit dem ST näherbringen. Dazu gibt es zunächst ein paar Grundlagen, denn viele können mit Ausdrücken wie z.B. 8 Bit Tiefe überhaupt nichts anfangen. Parallel dazu stellen wir Ihnen Software-Produkte der Firmen 3K Computerbild, Hofmann Engeneering und TmS vor. Hochauflösende Grafik ist angesagt!

Publishing Partner Master

Publishing Partner? Moment, das Programm kennen wir doch schon! Ja, richtig, schon im April 1987 haben wir dieses Programm testen können, damals in der Version 1.01. Doch seitdem sind einige Jahre ins Land gegangen. Haben Sie für das Programm etwas neues gebracht? Diese Frage wollen wir in der nächsten Ausgabe beantworten.

Zweite serielle Schnittstelle am ST

Sicher hat sich schon mancher ST-Anwender geärgert, daß sein ST nur eine RS-232-Schnittstelle hat - der TT dagegen gleich vier davon. Doch deshalb brauchen Sie nicht gleich auf den TT umzusteigen. Es läßt sich am ST über den MIDI-Port relativ einfach eine zweite serielle Schnittstelle (COM2) realisieren. Zwar muß man dabei in Kauf nehmen, daß der MIDI-Port nicht angesprochen werden kann, solange man die zweite RS-232 benutzt, aber wer spielt schon auf seinem Keyboard und macht zur gleichen Zeit DFÜ?

Mega-Tastatur am 1040ST

Es ist nun vier Jahre her, daß uns ATARI mit dem 1040STF beglückte. Leider wurde zu dieser Zeit die Tastatur vernachlässigt. Hier wurde sogar schon ein kleines Attentat auf die Fingerkuppen des Users gemacht. Nach zwei Jahren der Qual konnte man nun einen MEGA ST erwerben. Seine Vorteile bestehen unter anderem aus der hervorragenden, separaten Tastatur. Wir zeigen Ihnen, wie Sie sie an einen 1040ST anschließen können.

Die nächste ST Computer erscheint am Fr., den 31.8.90

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir weiterhin vorhaben, die Qualität zu steigern, haben wir Redakteure ein großes Anliegen an Sie, liebe Leserinnen und Leser:

Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur **Donnerstags von 14⁰⁰-17⁰⁰ Uhr** unter der Rufnummer 06196/481814 telefonisch beantwortet werden können. Fragen an den Versand unter 06196/481811.

Natürlich können wir Ihnen **keine** speziellen Einkauftips geben. Wenden Sie sich in diesem Fall bitte an einen Fachhändler. Wir können nur Fragen zur ST Computer beantworten.

Vielen Dank für Ihr Verständnis

Impressum ST Computer

Chefredakteur: Uwe Bärtels (UB)
Stellvertreter: Harald Egel (HE)

Redaktion:
Uwe Bärtels (UB)
Harald Egel (HE)
Joachim Merz (JM)
Martin Pittelkow (MP)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO), Claus Brod (CB), Ingo Brümmer (IB), Derek dela Fuente (ddF), Stefan Höhn (SH), Raymund Hofmann (RH), Dieter Kühner (DK), Claus P. Lippert (CPL), Markus Nerding (MN), Chr. Schormann (CS), R.Tolksdorf (RT), Thomas Werner (TW)

Autoren dieser Ausgabe:
M.Andrä, R.Blittkowsky, E.Böhnisch, D.Brockhaus, J.Decker, A.Esser, U.Hirschmann, A.Hollmann, M.Krusemark, P.Neuchel, R.Peiler, A.Rudolph, M.Schneider

Auslandskorrespondenz:
C.P.Lippert (Leitung), D.Dela Fuente (UK)

Redaktion: MAXON Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 14, FAX : 0 61 96/4 11 37

Verlag: Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57, FAX : 0 61 51/5 56 89 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Margaritis

Anzeigenpreise:
nach Preisliste Nr.5, gültig ab 1.3.90
ISSN 0932-0385

Layout:
Martin Lowack,
Manfred Zimmermann

Titelgestaltung:
Gunter Wenzel

Fotografie:
Martin Lowack

Illustration:
Manfred Zimmermann, Martin Lowack

Produktion:
K.H.Hoffmann, B.Kissner

Druck:
Frotscher Druck GmbH

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 8,-, ÖS 64,-, SFr 8,-
Jahresabonnement: DM 80,-
Europ. Ausland : DM 100,- Luftpost : DM 130,-
In den Preisen sind die gesetzliche MWSt. und die Zustellgebühren enthalten.

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit seiner Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern der MAXON Computer GmbH. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.


COMPUTERVIREN
Das neue
Anti Viren Kit 3.0
von G DATA
Jetzt auch für MS DOS*
und kompatible Computer:
ANTI VIREN KIT PC: DM 198,–**
Version für PC Speed: DM 99,–**
* MS DOS ist ein eingetragenes
Warenzeichen der Microsoft Corp
** Unverbindliche Preisempf
Kostenloser
Update-Service
für registrierte Benutzer.
Mit Harddisk-Überwachung.
Diagnose von Link-Viren.
Das meistverkaufte
Anti-Viren-Programm
für ATARI ST Computer
Anti Viren Kit 3.0
Anti Viren
DM 99,–
(unverbindl.
Preisempfehlung)
PROGRAMMABSTURZ!
DATENVERLUST!
Neue Version mit
Harddisk-Schreibüberwachung
und Virenscannung
Anti Viren Kit 3.0
bei Ihrem Atari-Händler
oder direkt bei
G DATA
Bundesrepublik:
Siemensstraße 16, D-4630 Bochum 1
Bestellungen: Telefon. 02 34/7 30 94
Telefax 02 34/7 30 83
Versand zzgl. DM 5,– Portopauschale
per Nachnahme oder Vorausscheck.
Schweiz:
Computertrend, Langstr. 31, CH-8021 Zürich
Österreich:
CSR Computer Rittner, Hauptstraße 34,
A-7000 Eisenstadt